2018

Improved embedded atom method potentials for metal hydride systems

Robert Fuller
lee.fuller455@gmail.com

IMPROVED EMBEDDED ATOM POTENTIALS FOR METAL HYDRIDE SYSTEMS

A thesis submitted to
the Graduate College of
Marshall University
in partial fulfillment of
the requirements for the degree of
Master of Science
in
Mechanical Engineering
by
Robert Fuller
Approved by
Dr. Iyad Hijazi, Committee Chairperson
Dr. Gang Chen
Dr. Ana Pena Alvarez

Marshall University
July 2018

APPROVAL OF THESIS

We, the faculty supervising the work of Robert Fuller, affirm that the thesis, "*Improved Embedded Atom Potentials for Metal Hydride Systems,*" meets the high academic standards for original scholarship and creative work established by the College of Information Technology and Engineering for the Master of Science in Mechanical Engineering. This work conforms to the editorial standards of our discipline and the Graduate College of Marshall University. With our signatures, we approve the manuscript for publication.

Dr. Iyad Hijazi, Weisberg Division of Engineering Committee Chairperson Date 6/7/2018

Dr. Gang Chen, Weisberg Division of Engineering Committee Member Date 6/7/2018

Dr. Ana Pena Alvarez, Weisberg Division of Engineering Committee Member Date 6/7/2018

ACKNOWLEDGMENTS

This work would not have been possible without the generous and insightful guidance of my advisor Dr. Iyad Hijazi. Special thanks also go to the NASA WV Space Grant Consortium. The Graduate Research Fellowship they awarded me was a tremendous aid to the completion of the research for this thesis.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Metal hydride systems are an important research topic in materials science because of their many practical, industrial, and scientific applications. Therefore, the development of reliable and efficient interatomic potentials for metal hydrides systems, to be utilized in molecular simulations, can be of great value in accelerating the research in this field. In this research, fully analytical interatomic Embedded Atom Method (EAM) potentials are developed for the PdAgH system. Ab initio simulations were performed to obtain the properties of selected PdAgH structures for fitting. The potentials are fit utilizing the central atom method without employing time-consuming molecular dynamics simulations in the fitting procedure. The new PdAgH potential extends a PdH model with fewer fitting parameters than previously developed EAM models for the hydride systems that can better predict the cohesive energy, lattice constant, bulk modulus, elastic constants, and the stable alloy crystal structures during molecular dynamics (MD) simulations for PdH over a wide range of hydrogen concentrations. MD simulation with the new PdAgH potential shows that the lattice constants and cohesive energies are in good agreement with the results from ab initio simulations for much of the hydrogen composition range. Simulations show that H atoms within the PdAg fcc structure move from tetrahedral positions to lower energy octahedral positions as predicted by the DFT results and previous works. Additionally, the elastic constants follow trends comparable to previous work and DFT calculations. Extension of the model to PdCuH and many other alloy combinations can now be easily accomplished building on this work.

CHAPTER 1

INTRODUCTION

Palladium–hydrogen systems have attracted a vast amount of research interest [1-8]. Its ability to readily absorb hydrogen at room temperature is reversible, and therefore can be applied in many applications, including fuel cells, hydrogen storage, refrigeration, catalytic converters, and nuclear radiation adsorption [9-15]. Furthermore, palladium has been widely utilized in hydrogen separation and purification to get an excellent combination of good H selectivity and permeability, nice mechanical properties, and high thermal stability [11-14]. However, the addition of hydrogen to palladium is made complex by the existence of a miscibility gap in the palladium-hydrogen system, which separates the material into low hydrogen (α) phase, and high hydrogen (β) phase [16] at room temperature.

There is a significant change in the lattice constant of the fcc lattice in passing from the α to the β phase [17]. The coexistence of the phases increases the risk of mechanical failure due to the strain caused by the lattice mismatch. Alloying palladium with other metals can increase its performance in many applications as well as substantially lowering the materials cost. Palladium silver alloys can offer increased H diffusivity and a less pronounced miscibility gap with much improved mechanical properties over pure palladium [18-23]. However, the relative insolubility of hydrogen in silver necessitates proper alloying of PdAg to obtain the best combination of properties [18].

The various industrial and technological applications of Pallidum Hydride have motivated interest in modeling this system. Extensive first-principles calculations have been carried out for PdH [24,25,14], and some have investigated PdAgH [26,27]. But while ab initio calculations are the most accurate simulation method, their high computational cost renders them

impractical for most system simulations containing more than a few hundred atoms. Molecular dynamics (MD) simulations utilizing empirical potentials such as the Embedded Atom Method (EAM) provide a more practical method for simulating more realistic system sizes containing a much larger number of atoms.

Although EAM potentials have been developed for some of the Pd-H systems, they were predominately designed for dilute hydrogen compositions, and therefore did not correctly estimate the miscibility gap [4]. Wolf et al. presented an EAM model for palladium hydrides that was utilized to predict the pressure-composition isotherm [17]. However, the authors did not provide the potential functions and the fitting parameters and did not test the mechanical properties. Zhou et al. developed an EAM potential model for the Pd-H system that was able to qualitatively predict the α and β phase miscibility gap [4]. Hale et al. later extended this model generating potentials for the PdAgH system [5]. Their fitting procedure used Mathematica in conjunction with MD simulation in LAMMPS. The model was able to produce smooth changes in the lattice parameter, cohesive energy and bulk modulus with varying hydrogen composition, as well as shifting hydrogen site occupancy and the disappearance of the miscibility gap at 300 K with increasing silver concentration. However, they constructed their high-order power series format Pd potential based on Foiles and Hoyt's Pd potential, available in tabular form but for which a complete description of the analytical formalism and the parameters are unavailable [28], which hinders further development of the palladium-silver hydride system.

In our recent work, Hijazi et al. have developed a new Pd-H potential that has fewer fitting parameters than previously developed EAM Pd-H potentials and is able to accurately predict the cohesive energy, lattice constant, bulk modulus, elastic constants, melting temperature, and the stable Pd-H structures in MD simulations with LAMMPS at various

hydrogen concentrations [29]. The potential also predicts the miscibility gap and preference for H occupancy at octahedral (OC) over tetrahedral (TE) interstitial sites.

In this thesis, the preceding Pd-H potential is extended to obtain a model for the PdAgH system. Ab initio calculations were performed using SIESTA to obtain fitting data for 14 different structures (7 OC and 7 TE structures) at four distinct hydrogen concentrations. Constrained nonlinear optimization is performed in MATLAB to minimize the error between the properties of the structure calculated and the fitting data obtained from experiment and first-principles calculations. The aim of this thesis is to introduce a fully analytical EAM potential for the PdAgH system, with fewer fitting parameters than previously developed models, which can predict the miscibility gap; preferred H occupancy sites; and accurately predicts the lattice constants, cohesive energies, and elastic properties in MD simulation. The central atom method is used, without the need for time-consuming MD simulations during the fitting procedure.

CHAPTER 2

MODELING

2.1 The Potentials

The Embedded Atom Method considers each atom as embedded in a host lattice consisting of all other atoms. The pair interaction between atoms is modeled as well as the energy associated with an atom's placement within the host lattice. With EAM the total energy of a system can be written as [30]

$$E_{tot} = \sum_{i=1}^{N} F_i(\rho_i) + \frac{1}{2}\sum_{i=1}^{N}\sum_{\substack{j=1 \\ j\neq i}}^{N} \varphi_{ij}(r_{ij}) \tag{1}$$

$$\rho_i = \sum_{j\neq i}^{N} f_j(r_{ij}) \tag{2}$$

where ρ_i is the electron density at atom i due to all other atoms and F_i is the energy to embed atom i in an electron density ρ_i. r_{ij} is the separation distance between atoms i and j and ϕ_{ij} is a two-body central potential between atoms i and j. f_j is the electron density of atom j as a function of distance from its center. Park and Hijazi have developed an empirical EAM potential for face-centered cubic (fcc) metals and alloys [31-35]. For the embedding function, they have taken the universal form given by

$$F(\rho) = F(\rho_e)\left[1-\eta\ln\left(\frac{\rho}{\rho_e}\right)\right]\left(\frac{\rho}{\rho_e}\right)^{\eta} \tag{3}$$

The host electron density is well represented with an exponential function having the following form:

$$f = f_e e^{-\chi(r-r_e)} \tag{4}$$

where f_e is a scaling constant given by the expression $f_e = E_c/\Omega$, where E_c is the cohesive energy and Ω is the atomic volume, r_e is the equilibrium nearest distance, and χ is a fitting constant. The pair potential is given by

$$\phi = -\phi_e\left[1+\delta(r/r_e-1)\right]e^{-\beta(r/r_e-1)} \tag{5}$$

which is the modified form of the potential function proposed by Rose et al. [36], and has three adjustable parameters, ϕ_e, δ, and β. In total, for fcc metals, we have six adjustable parameters χ, ϕ_e, δ, β, η, and ρ_e.

The structure of any binary hydrides and any ternary hydrides can be modeled with EAM, as the crystal structure is a metal lattice with hydrogen interstitials. For the H potential and the Pd-H and Ag-H potentials, the analytical expressions suggested by Zhou et al. will be utilized [4]. Hale et al. have developed an EAM potential for the ternary Pd-Ag-H system by extending the Pd-H potential of Zhou et al. [4,5]. In their implementation, the pair potential for hydrogen is given by

$$\phi_{HH}(r) = D_{HH}(\beta_{HH}e^{-\alpha_{HH}(r-r_{0,HH})} - \alpha_{HH}e^{\beta_{HH}(r-r_{0,HH})}) \tag{6}$$

where D_{HH}, α_{HH}, β_{HH}, and $r_{0,HH}$ are fitting constants, and $r_{0,HH}$ represents the equilibrium separation distance between two hydrogen atoms, and D_{HH} ($\beta_{HH} - \alpha_{HH}$) is its binding energy. The hydrogen electron density is approximated by the decreasing exponential function:

$$\rho_H(r) = C_H e^{-\delta_H \cdot r} \tag{7}$$

where C_H and δ_H are two fitting parameters. The hydrogen embedding function is expressed as

$$F_{H,u}(\rho) = -C_H \cdot \left(\frac{1}{2+d_H}\cdot\left(\rho+\varepsilon_H\right)^{2+d_H} - \frac{a_H+b_H}{1+d_H}\cdot\left(\rho+\varepsilon_H\right)^{1+dH} + \frac{a_H\cdot b_H}{d_H}\cdot\left(\rho+\varepsilon_H\right)^{d_H} \right) \tag{8}$$

where a_H, b_H, c_H, d_H, and $\varepsilon_H = 0.0540638$ are fitting parameters. The Pd-H and Ag-H pair

potentials have the form of the generalized Morse potential and is given by

$$\phi_{PdH}(r) = D_{PdH}(\beta_{PdH} e^{-\alpha_{PdH}(r-r_{0,PdH})} - \alpha_{PdH} e^{-\beta_{PdH}(r-r_{0,PdH})}) \tag{9}$$

$$\varphi_{AgH}(r) = D_{AgH}(\beta_{AgH} e^{-\alpha_{AgH}(r-r_{0,AgH})} - \alpha_{AgH} e^{-\beta_{AgH}(r-r_{0,AgH})}) \tag{10}$$

The EAM potential has a special property in that an elemental EAM potential is invariant to a transformation to the embedding energy function. The total energy is a linear expression of the embedding energy and pair functions, allowing them to be transformed according to

$$F_i^{Final}(\rho_i) = F_i^{initial}(\rho_i) + k\rho_i \tag{11}$$

$$\phi_{ij}^{Final}(r_{ij}) = \phi_{ij}^{initial}(r_{ij}) + 2k\rho_i(r_{ij}) \tag{12}$$

where *k* is an arbitrary constant. The Pd-H embedding and pair potentials were transformed in this manner following the methodology of Zhou et al. [4].

2.2 The Cohesive Energy

For a pure metal, the general format of cohesive energy is

$$E_c = \frac{1}{N}\left(\sum_{j=1}^{N} F_i(\rho_i) + \frac{1}{2}\sum_{j=1}^{N}\sum_{\substack{j=1 \\ j\neq i}}^{N} \phi_{ij}(r_{ij})\right) \tag{13}$$

To describe the binary alloy of host Pd and impurity H atoms, the EAM potential requires a total of seven functions for the two species: two embedding energy functions F_{Pd} and F_H, two atomic electron density functions f_{Pd} and f_H, and three pair potential energy functions $\phi_{Pd,}$ ϕ_H, and ϕ_{PdH}.

Utilizing these functions, the total cohesive energy of the Pd-H interstitial solid solution can be formulated as:

$$E_{coh} = \frac{1}{y+x}\left[\begin{array}{l} y\cdot\left(\sum_{i=1}^{N_{Pd}} F_{Pd,i}(\rho_{\mathrm{Pd},i}) + \frac{1}{2}\sum_{i=1}^{N_{Pd}}\sum_{\substack{j=1\\ j\neq i}}^{N_{Pd}} \phi_{PdPd}(r_{ij}) + \frac{1}{2}\sum_{i=1}^{N_{Pd}}\sum_{\substack{j=1\\ j\neq i}}^{N_{H}} \phi_{PdH}(r_{ij})\right) + \\ x\cdot\left(\sum_{i=1}^{N_{H}} F_{H,i}(\rho_{\mathrm{H},i}) + \frac{1}{2}\sum_{i=1}^{N_{H}}\sum_{\substack{j=1\\ j\neq i}}^{N_{H}} \phi_{HH}(r_{ij}) + \frac{1}{2}\sum_{i=1}^{N_{H}}\sum_{\substack{j=1\\ j\neq i}}^{N_{Pd}} \phi_{PdH}(r_{ij})\right) \end{array}\right] \tag{14}$$

The atomic electron densities are given by

$$\rho_{\mathrm{Pd},i} = \rho_{Pd-Pd,i} + \rho_{Pd-H,i} = \sum_{\substack{j=1\\ j\neq i}}^{N_{Pd}} f_{Pd}(r_{ij}) + \sum_{\substack{j=1\\ j\neq i}}^{N_{H}} f_{H}(r_{ij}) \tag{15}$$

$$\rho_{\mathrm{H},i} = \rho_{H-H,i} + \rho_{H-Pd,i} = \sum_{\substack{j=1\\ j\neq i}}^{N_{H}} f_{H}(r_{ij}) + \sum_{\substack{j=1\\ j\neq i}}^{N_{Pd}} f_{Pd}(r_{ij}) \tag{16}$$

2.3 Stress

It is well known that when a cubic lattice is homogeneously deformed by the application of an external force along the force direction, its state can be determined by its lattice parameters $\{a_i, i=1,2,\ldots,6\}$ where a_1, a_2, and a_3 are the length of the three edges of the unit cell of the cubic lattice; a_4, a_5, and a_6 are the three angles between the three edges. $\sigma_{H,\,ij}$ is the stress that can be evaluated using

$$\sigma_{H,ij} = \frac{1}{\Omega_o}\left[\frac{\partial F_H}{\partial \rho_{H,i}}\left(\frac{\partial \rho_{H-H,i}}{\partial r_{ij}} + \frac{\partial \rho_{H-I,i}}{\partial r_{ij}}\right) + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_H}\frac{\partial \phi_{H-H,ij}}{\partial r_{ij}} + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_I}\frac{\partial \phi_{H-I,ij}}{\partial r_{ij}}\right]\frac{r_i r_j}{r_{ij}} \tag{17}$$

2.4 Elastic Constants

For a unit cell at equilibrium the cohesive energy and elastic constants are related by the equation

$$E = E\big|_0 + \left.\frac{\partial E}{\partial a_i}\right|_0 u_i + \frac{1}{2}\left.\frac{\partial E}{\partial a_i a_k}\right|_0 u_i u_k + \ldots \tag{18}$$

where a_1, a_2 and a_3 are the edges lengths of the unit cell, and u_1, u_2 and u_3 are components of the displacement vector u. The elastic constants for a pure metal can be determined from (18) as follows [37]. For a homogeneous deformation, the nine strain parameters $\varepsilon_{ij} = \partial u_i / \partial a_j^e$ are constant, and $u_i = \varepsilon_{ij} a_j^e$. Thus for a homogeneous deformation,

$$E = A + A_{ij}\varepsilon_{ij} + \frac{1}{2}A_{ijkl}\varepsilon_{ij}\varepsilon_{kl} + \cdots \tag{19}$$

where

$$A = E\big|_0 \tag{20}$$

$$A_{ij} = \frac{1}{2}\left.\frac{\partial E}{\partial a_i}\right|_0 a_j^e \tag{21}$$

$$A_{ijkl} = \frac{1}{2}\left.\frac{\partial^2 E}{\partial a_i \partial a_k}\right|_0 a_j^e a_l^e \tag{22}$$

Classical elasticity theory for a system initially in equilibrium with no applied stress gives

$$E = -E_c + \frac{1}{2}\Omega_e C_{ijkl}\varepsilon_{ij}\varepsilon_{kl} + \cdots \tag{23}$$

Comparing Eq. (22) with (17) gives

$$A = -E_c \tag{24}$$

$$A_{ij} = 0 \tag{25}$$

$$A_{ijkl} = \Omega_e C_{ijkl} \tag{26}$$

where Ω_e is the volume per atom. The elastic constant C_{ijkl} can be determined by substituting Eqs. (14-16) into (26).

$$C_{ijkl} = \frac{1}{2\Omega_e}\frac{\partial^2 E}{\partial a_i \partial a_j} a_k a_l \tag{27}$$

The interatomic distance is related to the lattice distance by

$$r_{ij} = \sqrt{r_i^2 + r_j^2 + r_k^2} = \sqrt{(l_i a_i)^2 + (l_j a_j)^2 + (l_k a_k)^2}$$

$$C_{ijkl,u} = \frac{1}{2\Omega_e}\left[\left(\frac{r_i r_j}{r_{ij}^2}\right)\frac{\partial^2 E_u}{\partial r_{ij}^2} + \left(\delta_{il}\frac{1}{r_{ij}} - \frac{r_i r_j}{r_{ij}^3}\right)\frac{\partial E_u}{\partial r_{ij}}\right] r_j r_k \tag{28}$$

and for an interstitial solid solution we have

$$C_{ijkl,u} = \frac{1}{2\Omega_e}\left[\begin{array}{l} \dfrac{\partial^2 F_u}{\partial \rho_{u,i}^2}\left(V_{ij}^{u-H} + V_{ij}^{u-I}\right)\left(V_{kl}^{u-H} + V_{kl}^{u-I}\right) \\ + \dfrac{\partial F_u}{\partial \rho_{u,i}}\left(W_{ijkl}^{u-H} + W_{ijkl}^{u-I}\right) + B_{ijkl}^{u-H} + B_{ijkl}^{u-I} \end{array}\right] \tag{29}$$

where

$$W_{ijkl}^{u-H} = \left(\sum_{\substack{j=1 \\ j\neq i}}^{N_u}\frac{\partial^2 \rho_{u-H,i}}{\partial r_{ij}^2} - \sum_{\substack{j=1 \\ j\neq i}}^{N_u}\frac{1}{r_{ij}}\frac{\partial \rho_{u-H,i}}{\partial r_{ij}}\right)\frac{r_i r_j r_k r_l}{r_{ij}^2} + \delta_{ik}\frac{r_k r_l}{r_{ij}}\left(\frac{\partial \rho_{u-H,i}}{\partial r_{ij}}\right) \tag{30}$$

$$W_{ijkl}^{u-I} = \left(\sum_{\substack{j=1 \\ j\neq i}}^{N_u}\frac{\partial^2 \rho_{u-I,i}}{\partial r_{ij}^2} - \sum_{\substack{j=1 \\ j\neq i}}^{N_u}\frac{1}{r_{ij}}\frac{\partial \rho_{u-I,i}}{\partial r_{ij}}\right)\frac{r_i r_j r_k r_l}{r_{ij}^2} + \delta_{ik}\frac{r_k r_l}{r_{ij}}\left(\frac{\partial \rho_{u-I,i}}{\partial r_{ij}}\right) \tag{31}$$

$$B^{u-H}_{ijkl} = \frac{1}{2}\left[\sum_{\substack{j=1\\ j\neq i}}^{N_u}\frac{\partial^2\phi_{u-H,ij}}{\partial r_{ij}^2} - \frac{1}{r_{ij}}\sum_{\substack{j=1\\ j\neq i}}^{N_u}\frac{\partial\phi_{u-H,ij}}{\partial r_{ij}}\right]\frac{r_i r_j r_k r_l}{r_{ij}^2} + \frac{1}{2}\delta_{ik}\left(\sum_{\substack{j=1\\ j\neq i}}^{N_u}\frac{\partial\phi_{u-H,ij}}{\partial r_{ij}}\right)\frac{r_k r_l}{r_{ij}} \tag{32}$$

$$B^{u-I}_{ijkl} = \frac{1}{2}\left[\sum_{\substack{j=1\\ j\neq i}}^{N_u}\frac{\partial^2\phi_{u-I,ij}}{\partial r_{ij}^2} - \frac{1}{r_{ij}}\sum_{\substack{j=1\\ j\neq i}}^{N_u}\frac{\partial\phi_{u-I,ij}}{\partial r_{ij}}\right]\frac{r_i r_j r_k r_l}{r_{ij}^2} + \frac{1}{2}\delta_{ik}\left(\sum_{\substack{j=1\\ j\neq i}}^{N_u}\frac{\partial\phi_{u-I,ij}}{\partial r_{ij}}\right)\frac{r_k r_l}{r_{ij}} \tag{33}$$

$$V^{u-H}_{ij} = \left(\frac{\partial\rho_{u-H,i}}{\partial r_{ij}}\right)\frac{r_i r_j}{r_{ij}}, \qquad V^{u-I}_{ij} = \left(\frac{\partial\rho_{u-I,i}}{\partial r_{ij}}\right)\frac{r_i r_j}{r_{ij}} \tag{34}$$

$$V^{u-H}_{kl} = \left(\frac{\partial\rho_{u-H,i}}{\partial r_{ij}}\right)\frac{r_k r_l}{r_{ij}}, \qquad V^{u-I}_{kl} = \left(\frac{\partial\rho_{u-I,i}}{\partial r_{ij}}\right)\frac{r_k r_l}{r_{ij}} \tag{35}$$

Cubic crystals have three independent constants C_{11}, C_{12}, and C_{44} and we get the elasticity constants as follows: $C_{1111} = C_{11}$, $C_{1122} = C_{12}$, $C_{2323} = C_{44}$. In Eqs. (28-35) *u* indicates a central atom species. The elastic constants for the H_yI_x lattice structure are calculated by:

$$C_{ijkl} = \frac{y\cdot C_{ijkl,H} + x\cdot C_{ijkl,I}}{x+y} \tag{36}$$

2.5 Bulk Modulus

The bulk modulus for pure elements can be written as

$$B = \frac{a^2}{9V_o}\frac{\partial^2 E_c}{\partial a^2}, \tag{37}$$

we expanded as

$$B = \frac{a^2}{9V_o}\frac{\partial^2 E_c}{\partial a^2} = \frac{a^2}{9V_o}\frac{\partial^2 E_c}{\partial r_{ij}^2}\left(\frac{r_{ij}}{a}\right)^2 = \frac{a^2}{9V_o}\left[\frac{\partial^2 F}{\partial\rho^2}\left(\frac{\partial\rho}{\partial r_{ij}}\right)^2 + \frac{\partial F}{\partial\rho}\frac{\partial^2\rho}{\partial r_{ij}^2} + \frac{1}{2}\sum_{j\neq i}\frac{\partial^2\phi}{\partial r_{ij}^2}\right]\left(\frac{r_{ij}}{a}\right)^2 \tag{38}$$

For an interstitial solid solution with host and impurity atoms, the bulk modulus can be expended as:

$$B_u = \frac{a^2}{9V_o}\frac{\partial^2 E_u}{\partial a^2} = \frac{a^2}{9V_o}\frac{\partial^2 E_u}{\partial r_{ij}^2}\left(\frac{r_{ij}}{a}\right)^2 \tag{39}$$

$$B_u = \frac{a^2}{9V_o}\left[\frac{\partial^2 F_u}{\partial r_{ij}^2} + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_u}\frac{\partial^2 \phi_{u-H,ij}}{\partial r_{ij}^2} + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_u}\frac{\partial^2 \phi_{u-I,ij}}{\partial r_{ij}^2}\right]\left(\frac{r_{ij}}{a}\right)^2 \tag{40}$$

$$B_u = \frac{a^2}{9V_o}\left[\frac{\partial^2 F_u}{\partial \rho_{u,i}^2}\left(\frac{\partial \rho_{u,i}}{\partial r_{ij}}\right)^2 + \frac{\partial F_u}{\partial \rho_{u,i}}\frac{\partial^2 \rho_{u,i}}{\partial r_{ij}^2} + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_u}\frac{\partial^2 \phi_{u-H,ij}}{\partial r_{ij}^2} + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_u}\frac{\partial^2 \phi_{u-I,ij}}{\partial r_{ij}^2}\right]\left(\frac{r_{ij}}{a}\right)^2 \tag{41}$$

$$B_u = \frac{a^2}{9V_o}\left[\begin{array}{l}\frac{\partial^2 F_u}{\partial \rho_{u,i}^2}\left(\frac{\partial \rho_{u\text{-}H,i}}{\partial r_{ij}} + \frac{\partial \rho_{u\text{-}I,i}}{\partial r_{ij}}\right)^2 + \frac{\partial F_u}{\partial \rho_{u,i}}\left(\frac{\partial^2 \rho_{u-H,i}}{\partial r_{ij}^2} + \frac{\partial^2 \rho_{u-I,i}}{\partial r_{ij}^2}\right) + \\ \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_u}\frac{\partial^2 \varphi_{u-H,ij}}{\partial r_{ij}^2} + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_u}\frac{\partial^2 \varphi_{u-I,ij}}{\partial r_{ij}^2}\end{array}\right]\left(\frac{r_{ij}}{a}\right)^2 \tag{42}$$

where *u* indicates a central atom species, *H* and *I* referring to the host and impurity atoms respectively.

$$B_u = \frac{y \cdot B_H + x \cdot B_I}{x + y} \tag{43}$$

Alternatively, the three independent elastic constants C_{11}, C_{12}, and C_{44} of cubic crystals can represent the bulk modulus.

$$B = \frac{\left(C_{11} + 2 \cdot C_{12}\right)}{3} \tag{44}$$

2.6 Gibbs Free Energy of Mixing

The Gibbs free energy of mixing as a function of H concentration for PdH_x with $0 \leq x \leq 1$ is utilized to determine the phase miscibility gap and is given by

$$\Delta G^{mix} = \Delta H^{mix} - \Delta S^{mix} \cdot T \tag{45}$$

where ΔH_{mix} and ΔS_{mix} are the enthalpy and entropy of mixing and T is the temperature.

$$\Delta H^{mix} = E_{PdHx} - 2X \cdot E_{PdH} - (1-2X) \cdot E_{Pd} \tag{46}$$

where E_{PdHx}, E_{PdH}, E_{Pd}, are the cohesive energies, and X = x/(1+x) is the mole fraction.

$$\Delta S_t = -k_B \cdot \left[X \cdot \ln\left[X/(1-X) \right] + (1-2 \cdot X) \cdot \ln\left[(1-2X)/(1-X) \right] \right] \tag{47}$$

where k_B is Boltzmann's constant [4].

CHAPTER 3

PALLADIUM

3.1 Fitting

Utilizing equations (3), (4), and (5), the potential constants were obtained by fitting the lattice constant (a_0), the three elastic constants (C_{11}, C_{12}, C_{44}), the cohesive energy (E_c), and the vacancy formation energy (E_{vf}) of the pure metal *via* an optimization technique [30]. The fitting procedure was performed using a cutoff radius of r_{cut} = 5.35 Å. Table 1 lists the six fitting parameters for pure Pd. At temperature 0 K, (a_0) is the value of the lattice constant (a) in which the lattice is at equilibrium and the cohesive energy (E_c) is at the minimum. The cohesive energy was calculated from equation (13) while the lattice constant a_0 was determined from it's derivative, where $\partial Ec/\partial a = 0$ at the equilibrium lattice constant a_o. The elastic constants (C_{11}, C_{12}, C_{44}) were calculated utilizing equation (27). The unrelaxed vacancy formation energy (E_v^f) can be found from

$$E_v^f = -\frac{1}{2}\sum_m \phi(r_m) - \sum_m F(\bar{\rho}) + \sum_m F(\bar{\rho} - f(r_m)) \tag{48}$$

Table 1. Pd FCC fitting parameters

χ	ϕ_e	δ	β	η	ρ_e
2.054085	**0.216817**	**8.414105**	**7.221224**	**0.999999**	**3.316887**

Table 2. Pd properties from fitting results calculation

Element	**E_c (eV)**	**a_o (nm)**	**C_{11} (GPa)**	**C_{12} (GPa)**	**C_{44} (GPa)**	**B (GPa)**	**E_{vf} (eV)**
Calculation	**3.91**	**0.389**	**226.3**	**179.3**	**76.5**	**195**	**1.54**
Simulation	**3.91**	**0.389**	**226.3**	**179.3**	**76.5**	**195**	**-**
Foiles et al.	**3.91**	**0.389**	**239.2**	**173.5**	**65.6**	**195**	**1.58**
Experiment	**3.91**	**0.389**	**234.1**	**176.0**	**71.2**	**195**	**1.54**

3.2 Results

To validate the reliability of the obtained fitting parameters, the Pd EAM model was used in Molecular Dynamics (MD) Simulations, utilizing LAMMPS, to calculate the equilibrium cohesive energy, lattice constant, elastic constants, and bulk modulus. The MD results were in excellent agreement with the calculated values obtained at the end of the fitting procedure and in good agreement with the used experimental data. Table 2 summarizes the results obtained for Pd. The cohesive energy vs lattice constant was plotted against Rose et al.'s equation of state [36] and found to be in excellent agreement as shown in figure 1 below.

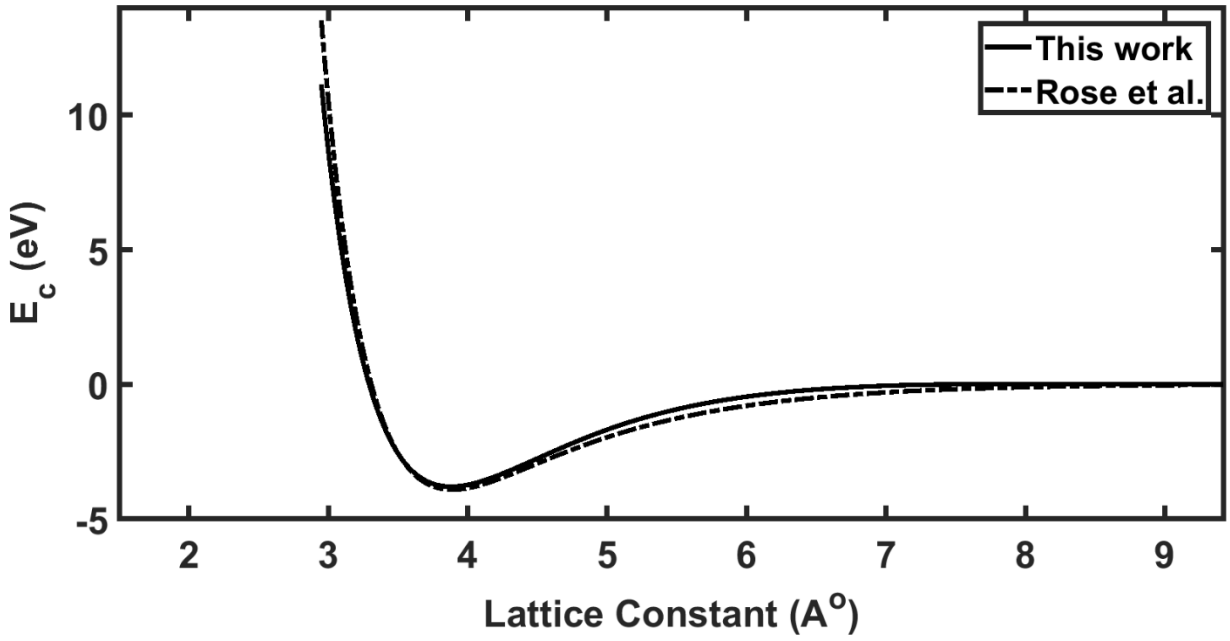

Figure 1. Pd equation of state.

The melting temperature of 1875 K calculated from MD simulation utilizing the Pd EAM model is in close agreement with the experimental value of 1828 K.

CHAPTER 4

PALLADIUM HYDRIDES

4.1 Fitting

In fitting the EAM PdH potential, among the properties to be considered is the cohesive energy, lattice constant, shear modulus $C' = (C_{11}-C_{12})/2$ and C_{44}, bulk modulus $B = (C_{11}+2C_{12})/3$, and Gibbs free energy of mixing [4]. Nonequilibrium tetrahedral (TE) and equilibrium octahedral (OC) PdH phase structures were included in the procedure. Figure 2 shows hydrogen OC and TE interstitial positions within a Pd lattice. The maximum composition for the PdH_x OC structure is $x = 1$ and for the TE structure is $x = 2$ [17]. In fitting the cohesive energy, the equilibrium structures must have the lowest cohesive energy.

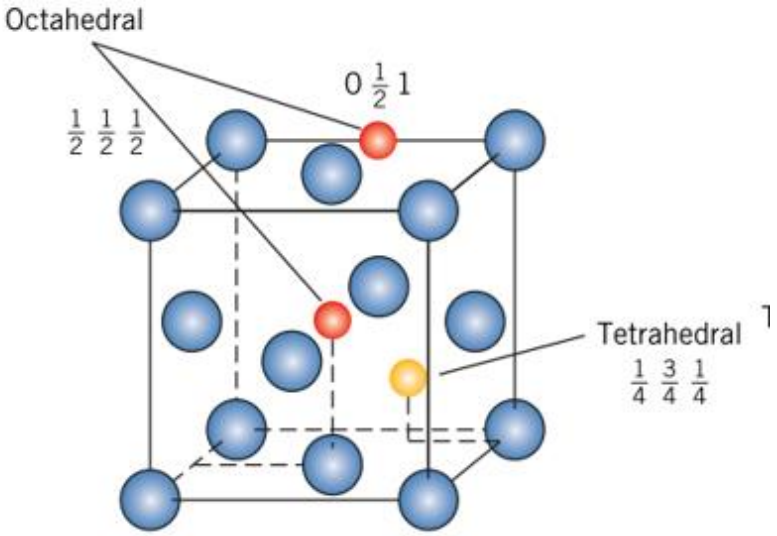

Figure 2. PdH structure with hydrogen atoms positions.

In this work, four OC PdH_x crystal structures with composition x = 0.25,0.5,0.75, and 1.0, two TE PdH_xH_x structures with x = 0.25 and 0.375, and an fcc H crystal were considered. The TE structures were included since they are less energetically favorable than the OC

structures with the same hydrogen concentration. The cohesive energy and the lattice constants for the six PdH structures were calculated using equation (14) respectively. Experimental values for the OC cohesive energies are available only for x = 0.5 and 1.0. As for the x = 0.25 and 0.75 structures, adjusted cohesive energies from first principles calculations were utilized [4]. The elastic constants C_{11} , C_{12} , and C_{44} were calculated using Eqs. (36) and fitted to the experimental data for the four OC crystal structures [38]. The Gibbs free energy of mixing as a function of hydrogen concentration was calculated using an equation derived by Zhou et al. [4]. The EAM potential parameter fitting was performed with constrained nonlinear optimization in MATLAB. Validation of the code is detailed in Appendix C. A cutoff radius of r_{cut} = 5.35 Å was used. The values from MATLAB calculations with the present model were in good to excellent agreement with the DFT and experimental data. Table 3 lists the fitting parameters for H-H and Pd-H obtained through our fitting procedure for the Palladium Hydride system.

Table 3. Fitting parameters for PdH

D_{HH}	α_{HH}	β_{HH}	C_H	δ_H	c_H	d_H
0.589510	**1.104827**	**0.942490**	**2.145808**	**0.942201**	**0.000100**	**1.187000**
D_{PdH}	α_{PdH}	β_{PdH}	a_H	b_H	$r_{0,PdH}$	$r_{0,HH}$
0.740938	**2.373944**	**1.702142**	**8.370790**	**62.343273**	**1.300000**	**3.474173**

4.2 Results

4.2.1 The Cohesive Energy

For PdH the shape of the cohesive energy as a function of H concentration is an important property as it influences the phase miscibility gap. All structures with hydrogen occupancy at OC interstitial sites should have lower cohesive energy at equilibrium than

structures with TE sites for the full range of H concentration. The cohesive energy results obtained from MD for $0 \leq x \leq 1$ composition are plotted in figure 3. The results are in good agreement with the experimental and DFT data utilized in the fitting procedure, better agreement than those obtained by Zhou et al. [4].

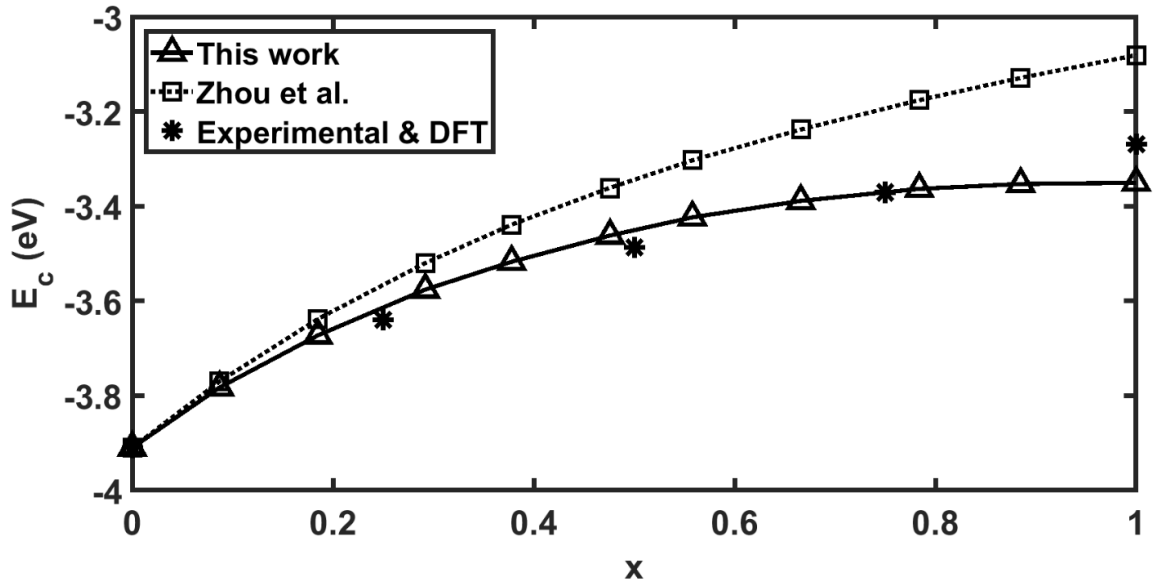

Figure 3. PdH_x cohesive energy from MD, Zhou et al., and Fitting Data.

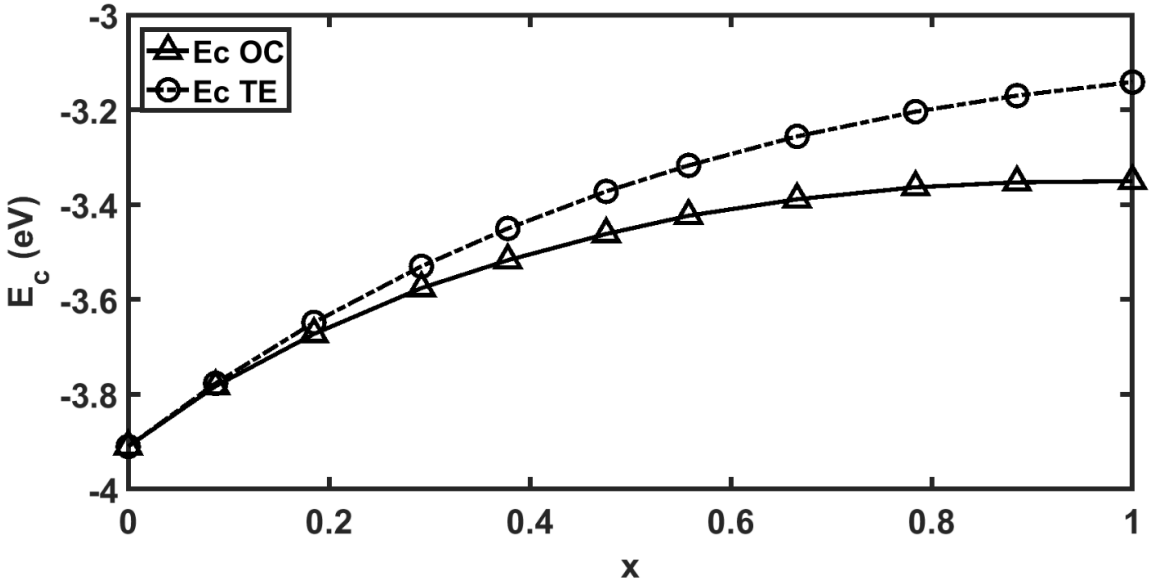

Figure 4. PdH_x OC & . PdH_x TE cohesive energy.

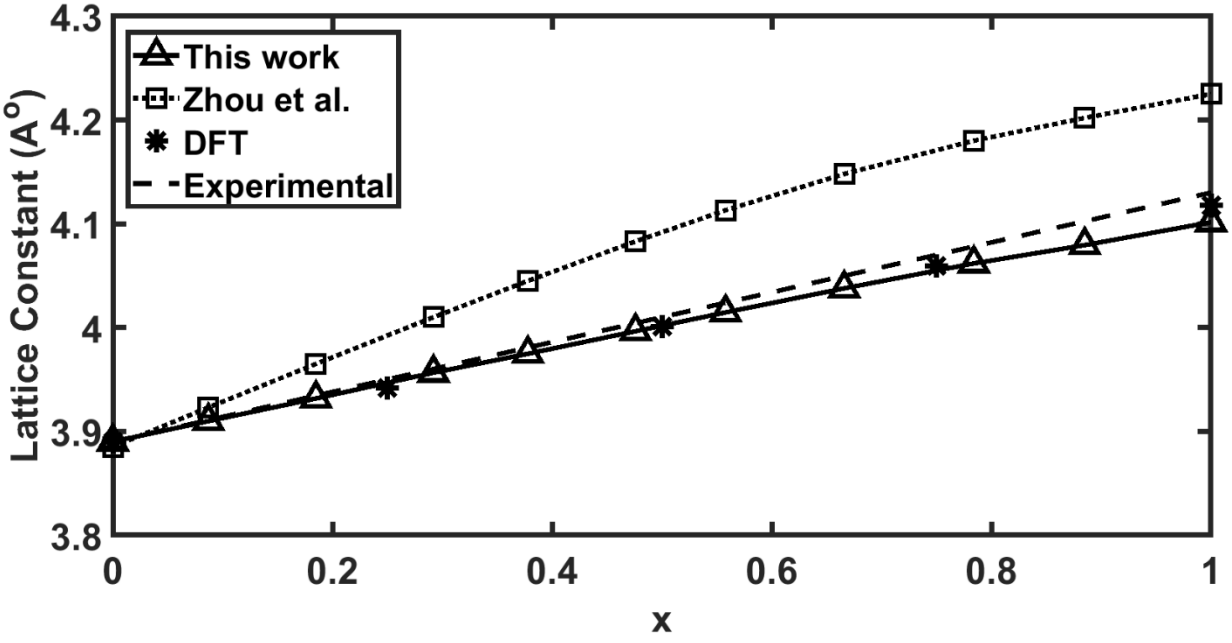

Figure 5. PdH_x Lattice Constants from MD, Zhou et al., Fitting Data, and Experiment.

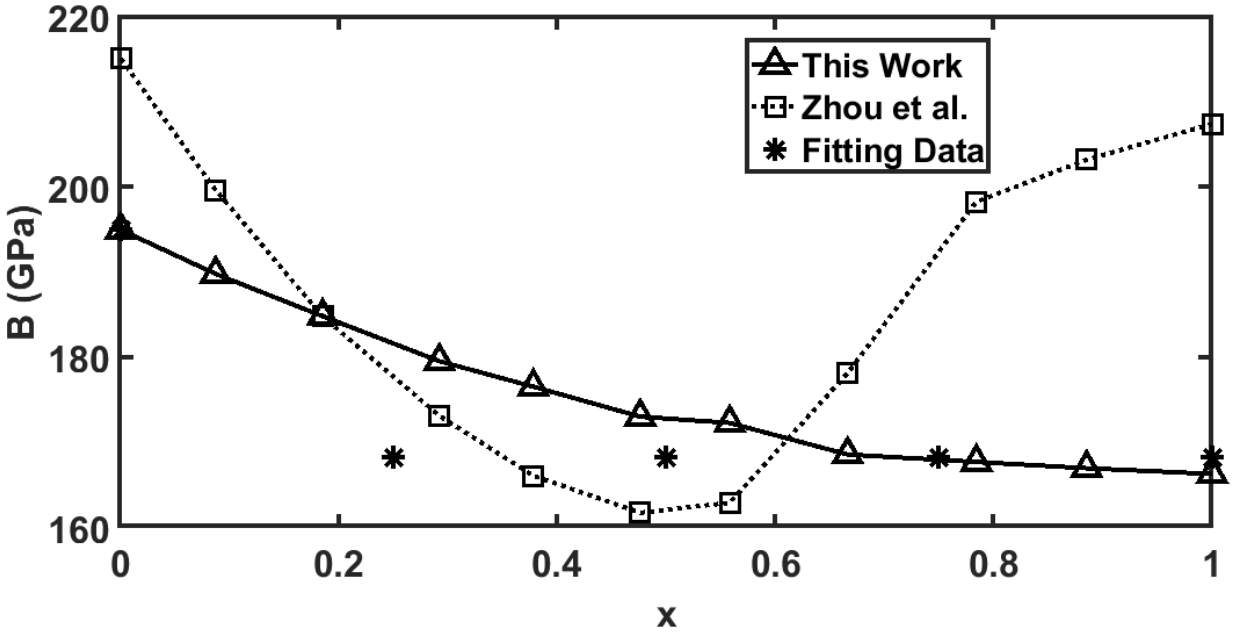

Figure 6. PdH_x Bulk Modulus from MD, Zhou et al., and fitting data lower limit.

In figure 4 the calculated cohesive energies as functions of H concentration are plotted for the TE and OC structures. The cohesive energies of the OC structures are the energies obtained by first conducting MD simulations with annealing, from 500 K to 0 K in 100 ns with a Nose-Hoover NPT ensemble. Each MD simulation was then followed by a molecular static (MS) simulation, utilizing the conjugate gradient (cg) minimization procedure. The same procedure cannot be used for the PdH TE interstitial structures as all hydrogen atoms will diffuse to the more stable OC interstitial sites at the end of the simulations [4]. Therefore, the cohesive energy for the TE structures was determined using Eq. (14) at the equilibrium lattice constant a_0.

4.2.2 Equilibrium Lattice Spacing

An important property is the PdH lattice constant at equilibrium as a function of H concentration, as it induces stress when neighboring regions have different concentrations [4]. To evaluate the present potential, the lattice constant for the OC PdH_x structure as a function of composition ($0 \leq x \leq 1$) was obtained from MD simulations at equilibrium and plotted in figure 5. For comparison, the DFT data used in the fitting procedure and those of Zhou et al. are also plotted in figure 5. The DFT values show an increasing trend with the hydrogen composition. From the plot, it is evident that our equilibrium lattice constant curve from the MD simulations is in better agreement with the DFT and experimental data [39] than those obtained by Zhou et al. [4].

4.2.3 Dynamic Stability

In palladium hydride, H atoms prefer to occupy OC interstitial positions in the Pd fcc structure [40]. To test the structural stability for PdH_x structures with our EAM model we used LAMMPS to create various fcc structures with H atoms occupying TE positions as shown in figure 7(a). Using MD simulation with an NPT ensemble each TE structure was annealed for 100

ns from 500 K to 1 K. At the end of each MD + MS simulation, the hydrogen atoms moved to the lower energy OC interstitial positions as shown in figure 7(b).

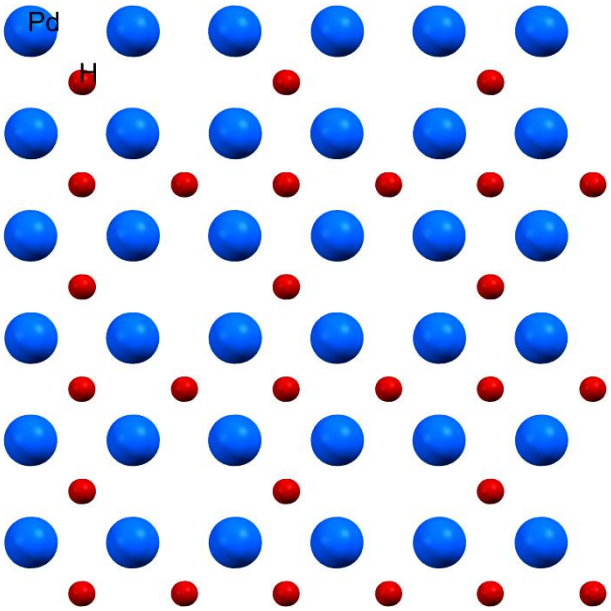

Figure 7(a) PdH TE structure before simulation.

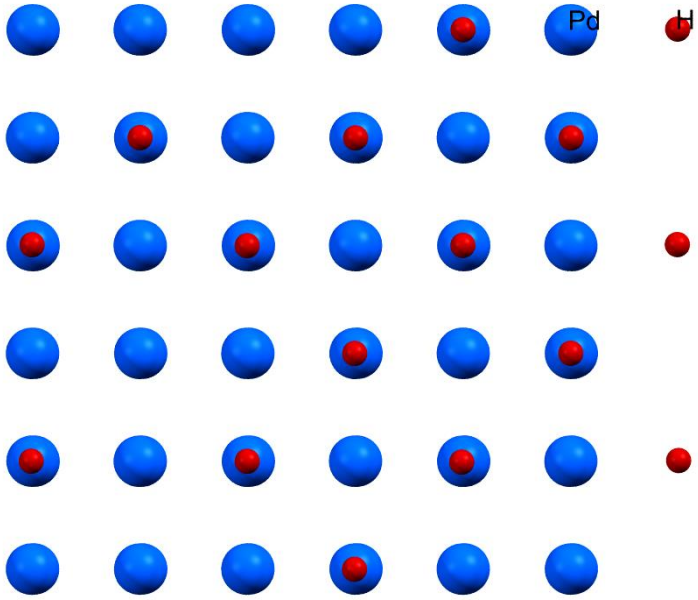

Figure 7(b) PdH OC structure after simulation.

4.2.4 The Miscibility Gap

The dilute α and concentrated β phases and the phase miscibility, where the single-phase alloy is less stable than the two-phase alloy, can be determined from the curve of the Gibbs free energy of mixing as a function of PdH_x concentration where the Gibbs energy plot shows multiple minima at equilibrium [24,41]. To validate that the present EAM potential can predict the miscibility gap, the Gibbs free energy for the Pd-H octahedral structures was calculated utilizing equation (45) with the cohesive energy values from MD simulations. Figure 8 shows the results at 0 K, 300 K, and 500 K. At 0 K, the Gibbs free energy is positive for all compositions, implying that the attractive interactions between atoms of different species are, on average, weaker than those between molecules of the same species. At room temperature, experimental results indicate that the mole fractions of the α and β phases are at 0.03 and 0.375 [42]. At 300 K, a minimum can be seen developing at low concentration consistent with experiment, however the width of the miscibility gap is overpredicted as the β phase mole fraction is 0.50. As the temperature increases, the miscibility gap becomes more defined. The Gibbs free energy of mixing curve at 500 K displays two minima, approximately at the equilibrium mole fractions X = 0.08 and 0.40. These two minima are the α and β phases and define the miscibility gap region. For the potential of Zhou et al., the mole fractions at 500K for the α and β phases are approximately 0.1 and 0.4. Therefore, the present EAM model is capable of producing the miscibility gap and is in good agreement with that of Zhou et al. [4] at 500 K.

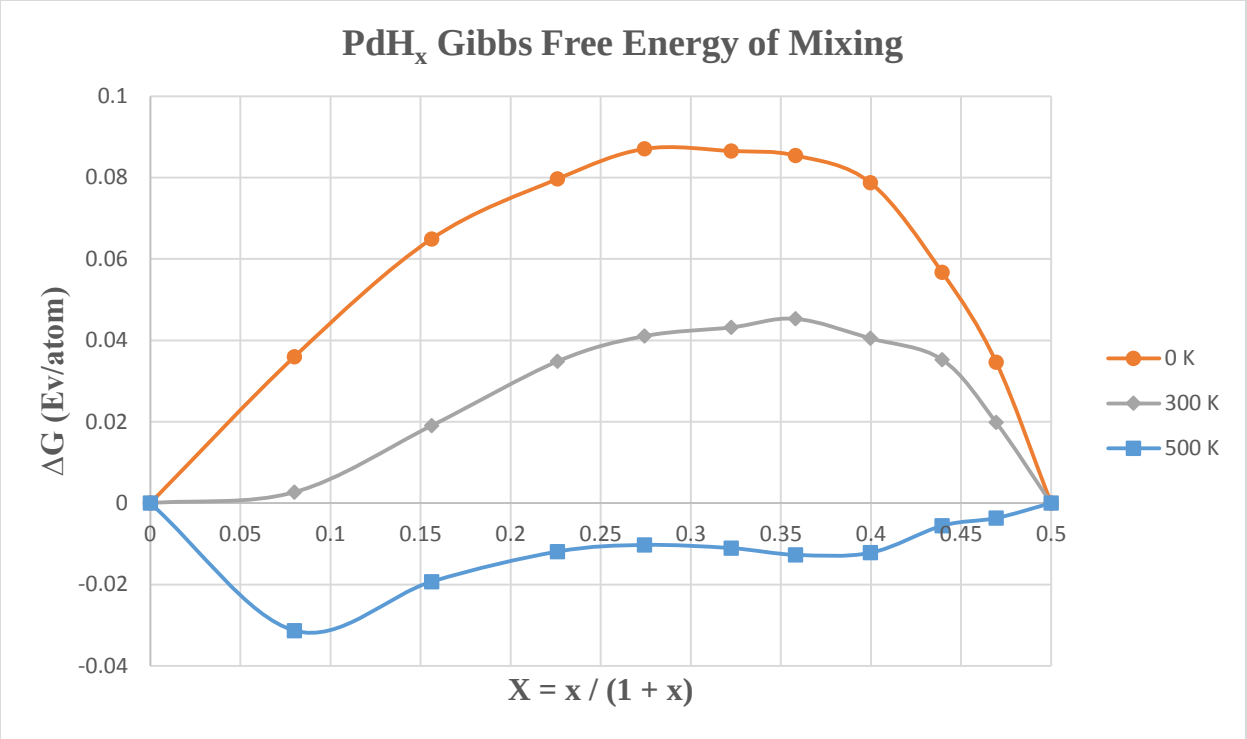

Figure 8. Gibbs free energy of mixing associated with hydrogen composition.

4.2.5 The Elastic Constants and Bulk Modulus

To test the validity of our PdH model we ran Static Molecular simulations utilizing LAMMPS to calculate the three lattice constants C_{11}, C_{12}, and C_{44} for a set of PdH_x lattice structures with hydrogen atoms at octahedral sites. The elastic constants were estimated utilizing a general-purpose script written by Aiden P. Thompson at Sandia Laboratories [43].

Table 4. PdH properties from MD simulations

H/Pd	E_c (eV)	a (Å)	C_{11} (GPa)	C_{12} (GPa)	C_{44} (GPa)	B (GPa)	C' (GPa)
0.000	**3.91**	**3.89**	**226.31**	**179.35**	**76.55**	**195.01**	**23.48**
0.087	3.78	3.91	221.34	174.06	74.21	189.82	23.64
0.185	3.66	3.93	216.49	169.04	71.93	184.86	23.72
0.250	**3.59**	**3.94**	**213.78**	**166.26**	**70.57**	**182.10**	**23.76**
0.292	3.56	3.95	211.63	163.56	68.73	179.58	24.035
0.378	3.51	3.97	207.94	160.92	65.57	176.59	23.51
0.476	3.45	3.99	203.53	157.83	62.00	173.07	22.85
0.500	**3.41**	**4.00**	**197.46**	**155.70**	**61.54**	**169.62**	**20.88**
0.558	3.40	4.01	197.82	159.58	63.81	172.33	19.12
0.666	3.38	4.03	200.57	152.64	59.78	168.63	23.96
0.750	**3.36**	**4.05**	**201.21**	**151.70**	**62.21**	**168.21**	**24.75**
0.784	3.36	4.06	200.38	151.43	61.48	167.75	24.47
0.885	3.36	4.08	199.78	150.63	60.33	167.02	24.57
1.000	**3.35**	**4.10**	**200.81**	**149.11**	**60.37**	**166.34**	**25.85**

While Zhou et al. results indicate that the bulk modulus first decreases and then increases as the hydrogen concentration increases, our results from MD indicates that the bulk modulus strongly decreases as the concentration increases in the Pd fcc structure. These results imply that the material gets softer and becomes susceptible to deformation as the hydrogen concentration increases. The observation that the bulk modulus and strength decrease with increasing hydrogen concentration was also previously reported by Zhong et al. [44], and recently by Niranjan Ilawe et al [24].

CHAPTER 5

PALLADIUM SILVER ALLOYS

5.1 Fitting

An Ag EAM potential was fitted in the same manner as was performed for Pd. The fitting parameters are given in table 5. The predicted properties from fitting calculations were nearly identical to the simulation results and in excellent agreement with the experimental data as shown in table 6. Additionally, the equation of state was in excellent agreement with that of Rose et al. [36] as can be seen in figure 9.

Table 5. Ag and Pd-Ag fitting parameters

χ	ϕ_e	δ	β	η	ρ_e	S_{Ag}	S_{Pd}
1.584768	**0.154164**	**8.491335**	**7.183185**	**1.022270**	**2.213230**	**1.8319**	**1.1063**

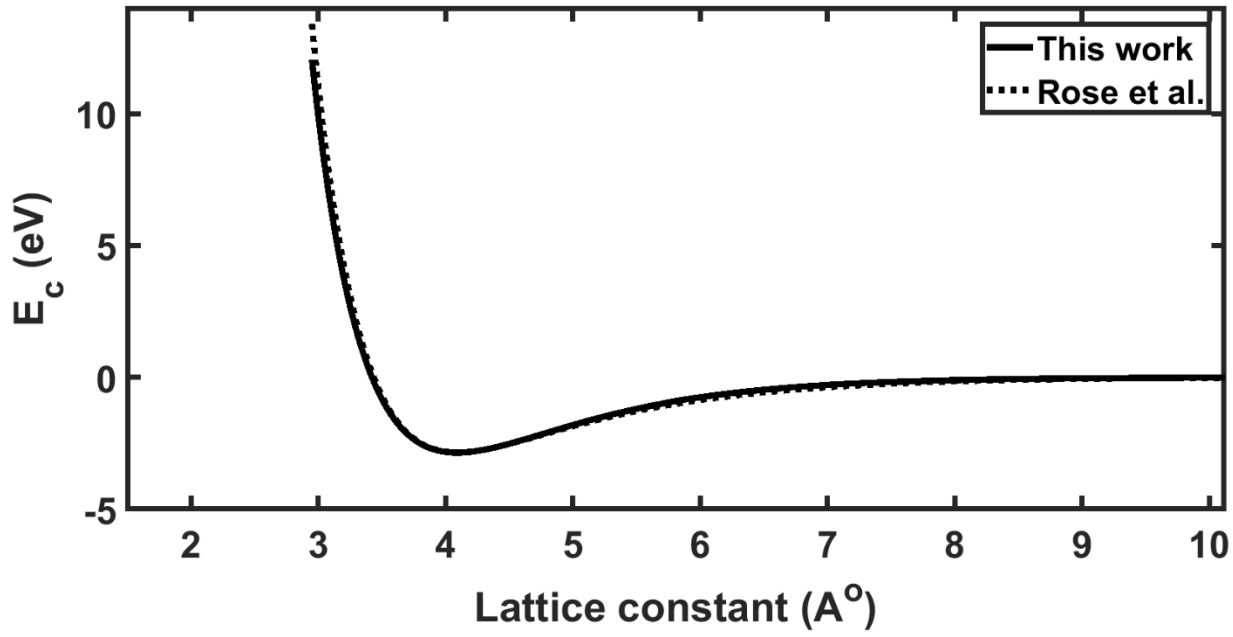

Figure 9. Ag equation of state.

Table 6. Ag properties from fitting results calculation

Element	E_c (eV)	a_o (nm)	C_{11} (GPa)	C_{12} (GPa)	C_{44} (GPa)	B (GPa)	E_v^f (eV)
This work	**2.85**	**0.409**	**123.1**	**94.4**	**46.9**	**104**	**1.08**
Simulation	**2.85**	**0.409**	**123.1**	**94.4**	**46.9**	**104**	**-**
Foiles et al.	**2.85**	**0.409**	**124**	**91**	**57**	**-**	**-**
Experimental	**2.85**	**0.409**	**124**	**93.4**	**46.1**	**-**	**1.10**

Having obtained reliable Pd and Ag potentials, a cross pair two-body potential must be specified to model Pd-Ag alloys. The mixing rule developed by Johnson [45] was used; it incorporates the electron densities and pair potential functions of the interacting type a and type b atoms:

$$\phi_{ab}(r) = \frac{1}{2}\left[\frac{f_b(r)}{f_a(r)}\phi_{aa}(r) + \frac{f_a(r)}{f_b(r)}\phi_{bb}(r)\right] \quad (49)$$

The adjustable atomic electron density parameter for each alloy metal is determined by the relationship $f_e = S(E_c/\Omega)$, where S is a fitting parameter to be determined by the dilute-limit heats of solution, E_c is the cohesive energy and Ω is the atomic volume. For pure metals, $S = 1$. The dilute-limit of the unrelaxed heat of solution for atom type-b as an impurity or solute and atom type-a as the host or solvent was calculated using the following five steps:

(a) Remove host: (b) Add impurity:

$$F^b(\rho^{-a}) + \sum_{i\neq 1}\phi^{ab}\left(r_{li}^a\right) \qquad -F^a(\rho^{-a}) - \sum_{i\neq 1}\phi^{aa}\left(r_{li}^a\right)$$

(c) Adjust neighbors:

$$-\sum_{i\neq 1}F^a(\rho^{-a}) - \sum_{i\neq 1}F_i^a(\rho^{-a} - f^a\left(r_{li}^a\right) + f^b\left(r_{li}^a\right))$$

(d) Adjust cohesive energy: (e) Relaxation energy:

$$-E_c^a + E_c^b \qquad E_r = \left[1.167(\Omega_b/\Omega_a - 1)\right]^2$$

where ρ^a is the equilibrium electron density at an atom site in a perfect crystal of a-type atoms and r^a is the equilibrium nearest-neighbor distance in this perfect crystal. E_r is the decrease in energy due to relaxation, as calculated by Johnson [45], and is predominantly dependent on the size mismatch. The subscripts a and b indicate host and impurity, respectively.

The electron density scaling parameters S_{Pd} and S_{Ag} obtained from fitting are given in table 5. The heat of solution calculated with each metal are given in table 7, both with and without the approximation for relaxation of the lattice. The values are in better agreement with experiment than those of both Foiles et al. [46] and Hijazi and Park [31].

Table 7. Pd-Ag heat of solution from fitting results calculation

$\Delta H_{solution}$ (eV)	This work	This work (relaxation)	Experimental	Foiles et al.	Hijazi & Park
Pd in Ag	**-0.150**	**-0.186**	**-0.11**	**-0.36**	**-0.19**
Ag in Pd	**-0.180**	**-0.207**	**-0.29**	**-0.24**	**-0.14**

5.2 Results

The fitted Pd-Ag potential was used to generate a tabulated potential file in eam.alloy DYNAMO *setfl* format utilizing a MATLAB code written for this purpose. Validation of this code is discussed in Appendix C. MD simulations were performed for the PdAg structures using the generated potential file with LAMMPS. The MD simulations further demonstrated the reliability of the Pd-Ag potentials; the results are shown in figures 10-15. The lattice constants and cohesive energies of the Pd_xAg_{1-x} alloys were in excellent agreement with the experimental data throughout the composition range $0 \leq x \leq 1$. The elastic constants C_{12}, C_{11}, and C' showed good agreement with the trend predicted by DFT calculations [47]. The bulk modulus also matches the trends given by DFT calculations [47,26] and very accurately match the experimental values available which are at the edge of the composition range.

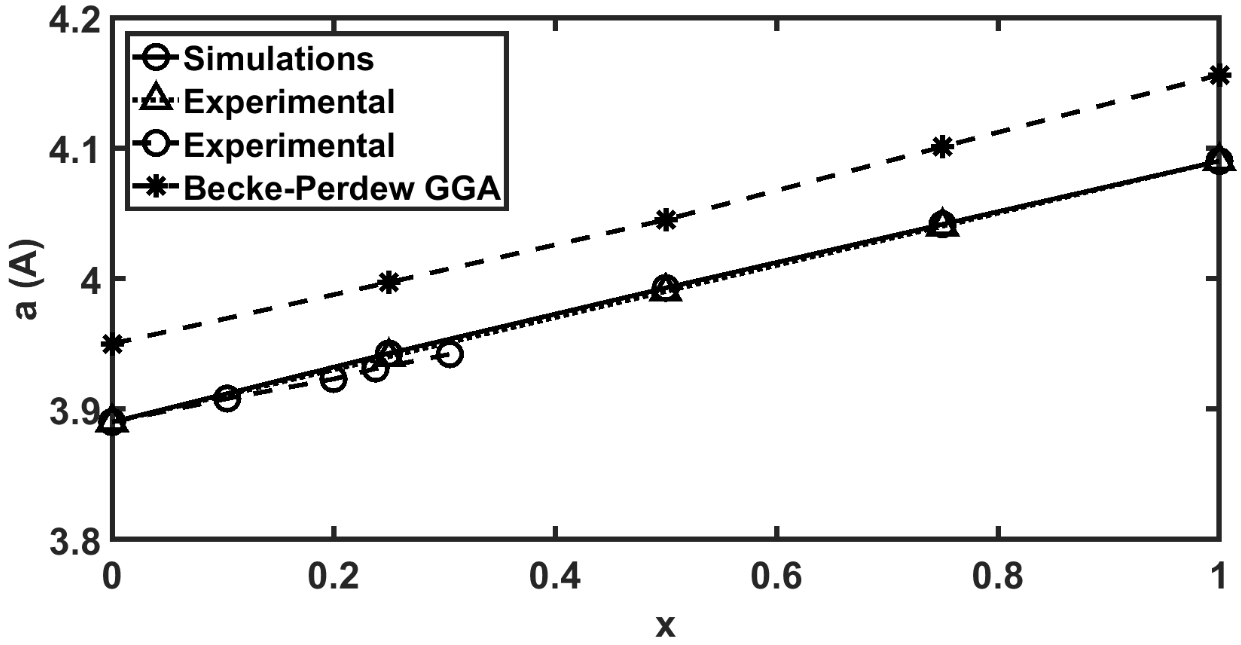

Figure 10. $Pd_{1-x}Ag_x$ alloys lattice constants from MD, Experiments, and DFT [20].

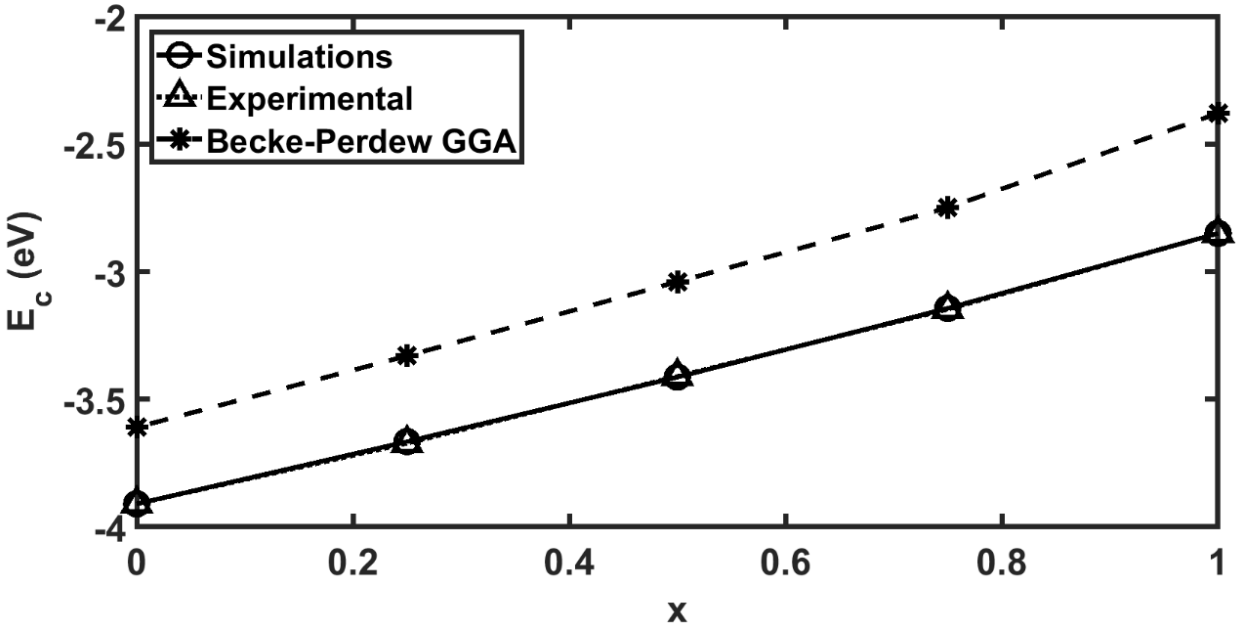

Figure 11. $Pd_{1-x}Ag_x$ alloys cohesive energies from MD, Experiment, and DFT [21].

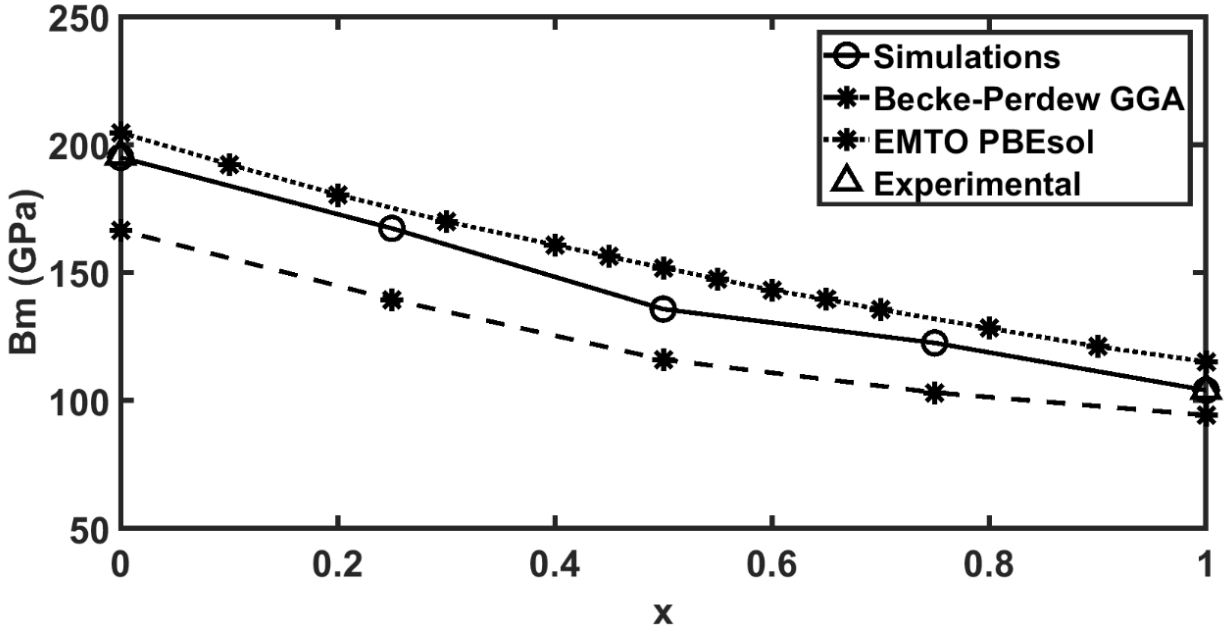

Figure 12. $Pd_{1-x}Ag_x$ alloys bulk modulus from MD, Experiment, and DFT [20, 21].

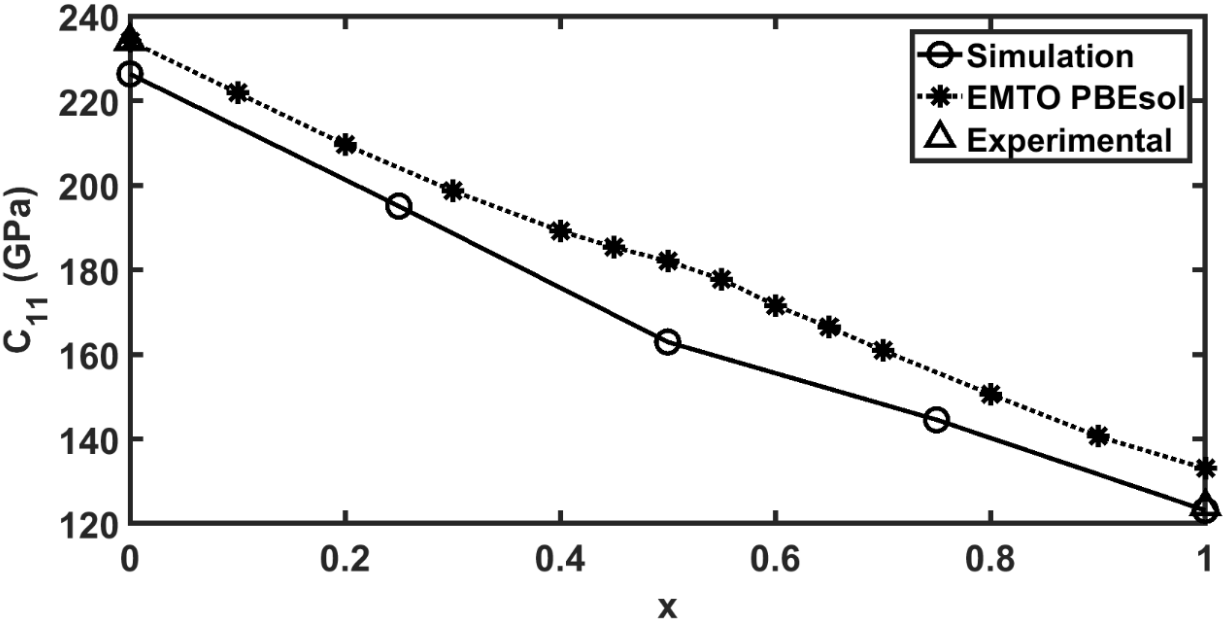

Figure 13. $Pd_{1-x}Ag_x$ alloys C_{11} elastic constant from MD, Experiment, and DFT [20].

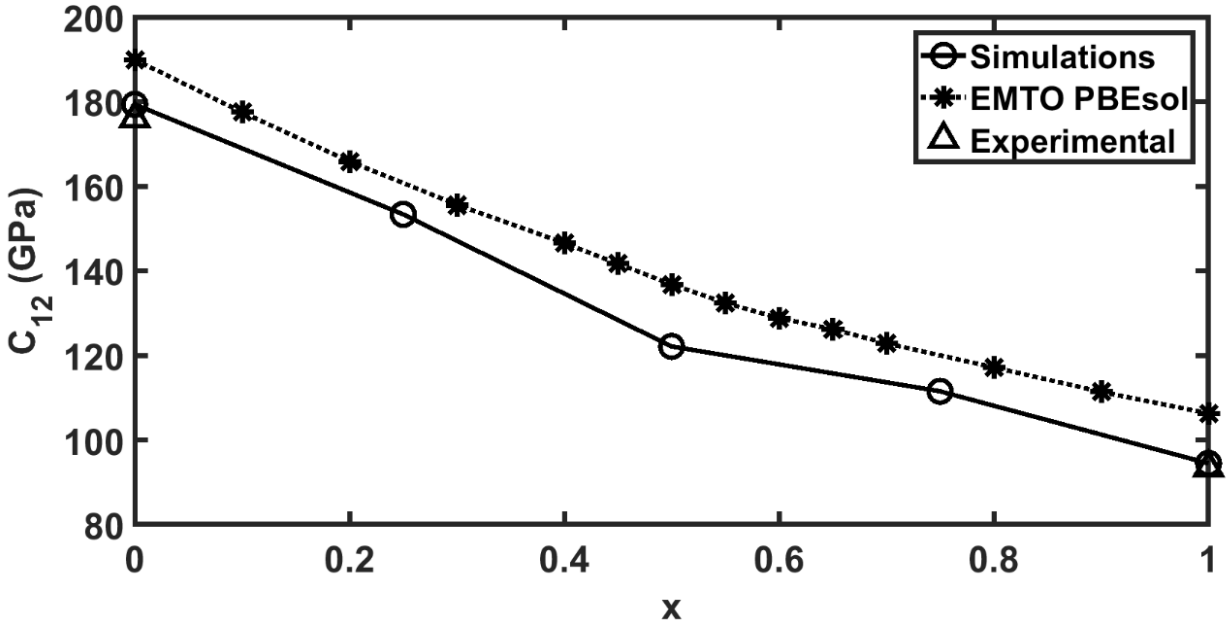

Figure 14. $Pd_{1-x}Ag_x$ alloys C_{12} elastic constant from MD, Experiment, and DFT [20].

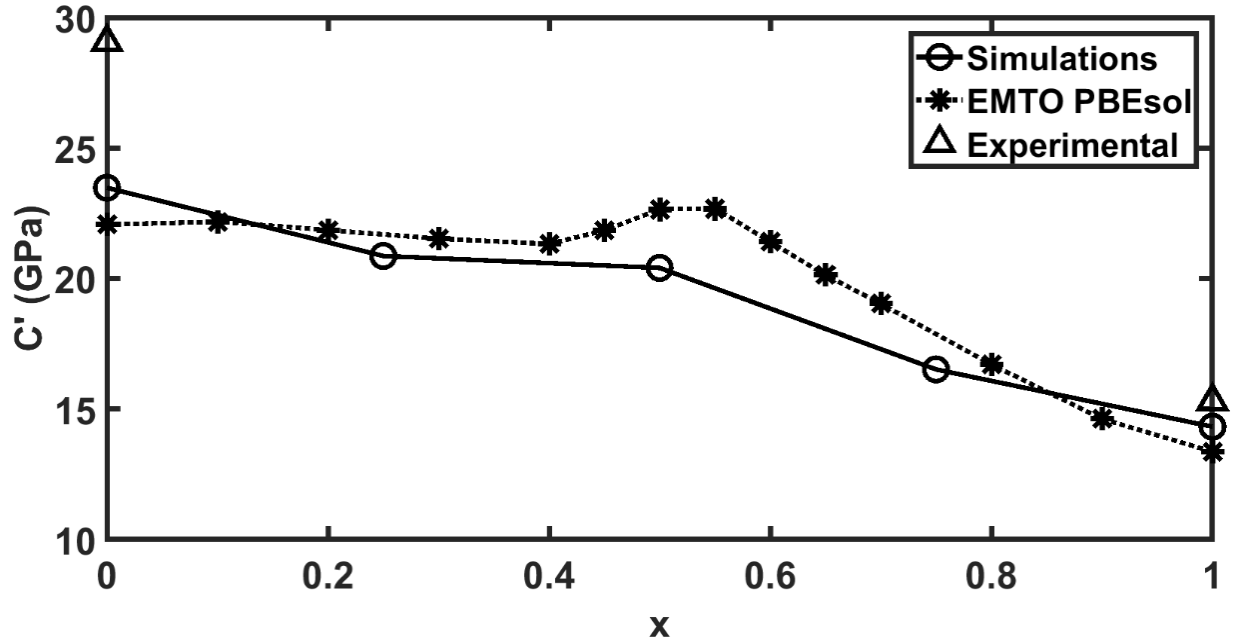

Figure 15. $Pd_{1-x}Ag_x$ alloys C' elastic constant from MD, Experiment, and DFT [20].

CHAPTER 6

PALLADIUM SILVER HYDRIDES

6.1 DFT Calculations

Data is needed for fitting the φ_{Ag-H} pair function. Because hydrogen is poorly soluble in silver [48], there is not experimental AgH data available for fitting in the same manner as was done for PdH. Consequently, PdAgH properties must be used for fitting. There is little experimental data available and few ab initio studies have investigated a sufficient range of hydrogen composition to be used for fitting. Hale et al. performed their own DFT calculations to obtain fitting data [5]; however, they neglected to report the lattice constants of the structures; only the cohesive energies were given.

In this work, we obtained data for PdAgH structures by performing DFT calculations using the open source SIESTA program. The DFT results are based on local density approximation (LDA) methods using projector-augmented wave (PAW) pseudopotentials. A single unit cell with three Pd atoms, one Ag atom, and one to four H atoms at various positions was used for the calculations. The calculations utilized a cutoff energy of 20 Ry, a dense $18\times18\times18$ Monkhorst–Pack grid, an electronic temperature of 25 K, and account for electron spin polarization.

The $Pd_{0.75}Ag_{0.25}H_y$ structures were investigated for y = 0, 0.25, 0.50, 0.75, 1.00. Hydrogen can occupy three distinct interstitial sites within the PdAg FCC lattice. As was the case for PdH, these include octahedral positions, both body center (O1) and edge (O2), and tetrahedral positions. These sites are illustrated in figure 16 [5].

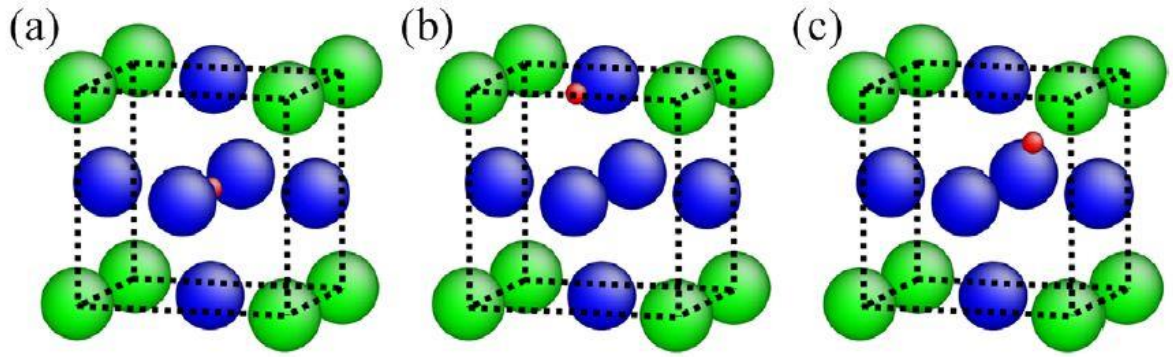

Figure 16. H (red) interstitial sites within the Pd (blue) and Ag (green) lattice.

As is typical of DFT, the simulations reliably overpredicted the cohesive energies and lattice constants of the structures. The results of the calculations can be shifted by multiplication with a factor, selected for each element present, so that the results for the Pd, $PdH_{0.50}$, $PdH_{1.00}$, and $Pd_{0.75}Ag_{0.25}$ structures match the experimentally determined values. The following equations describe the shifting procedure for the cohesive energy. The procedure is the same for the lattice constant. The shifting data is given in table 8 and the shifting factors in table 9.

$$\left(\text{Cohesive Energy}_{\text{Pd}_{0.75}\text{Ag}_{0.25}\text{H}_x}\right)_{SIESTA\ shifted} = \left(\text{C}_{\text{Pd}_{0.75}\text{Ag}_{0.25}} + \text{x} \cdot \text{C}_{\text{H}}\right) \cdot \left(\text{Cohesive Energy}_{\text{Pd}_{0.75}\text{Ag}_{0.25}\text{H}_x}\right)_{SIESTA} \quad (50)$$

$$\text{C}_{\text{Pd}_{0.75}\text{Ag}_{0.25}} = \frac{\left(\text{Cohesive Energy}_{\text{Pd}_{0.75}\text{Ag}_{0.25}}\right)_{Experimental}}{\left(\text{Cohesive Energy}_{\text{Pd}_{0.75}\text{Ag}_{0.25}}\right)_{SIESTA}} \quad (51)$$

$$\text{C}_{H} = 2 \cdot \left\{ \frac{\left(\text{Cohesive Energy}_{\text{PdH}_{0.50}}\right)_{Experimental}}{\left(\text{Cohesive Energy}_{\text{PdH}_{0.50}}\right)_{SIESTA}} - \text{C}_{Pd} \right\} \quad (52)$$

$$\text{C}_{Pd} = \frac{\left(\text{Cohesive Energy}_{\text{Pd}}\right)_{Experimental}}{\left(\text{Cohesive Energy}_{\text{Pd}}\right)_{SIESTA}} \quad (53)$$

Table 8. Experimental values used in shifting ab initio data

	Lattice Constant (Å)	Cohesive Energy (eV)
Pd	**3.89**	**-3.91**
$PdH_{0.50}$	**-**	**-3.4877**
$PdH_{1.00}$	**4.12**	**-**
$Pd_{0.75}Ag_{0.25}$	**3.94**	**-3.65**

Table 9. PdAgH ab initio data shifting factors

Lattice Constant Shifting Factors		Cohesive Energy Shifting Factors	
$C_{Pd_{0.75}Ag_{0.25}}$	0.98546	$C_{Pd_{0.75}Ag_{0.25}}$	0.78677
C_H	-0.017677	C_H	-0.021070
C_{Pd}	0.98232	C_{Pd}	0.79165

The values before and after shifting are given in Table 8. The results are in fair agreement with those of Hale et al. [5], matching the trend over the various structures as shown by figure 16.

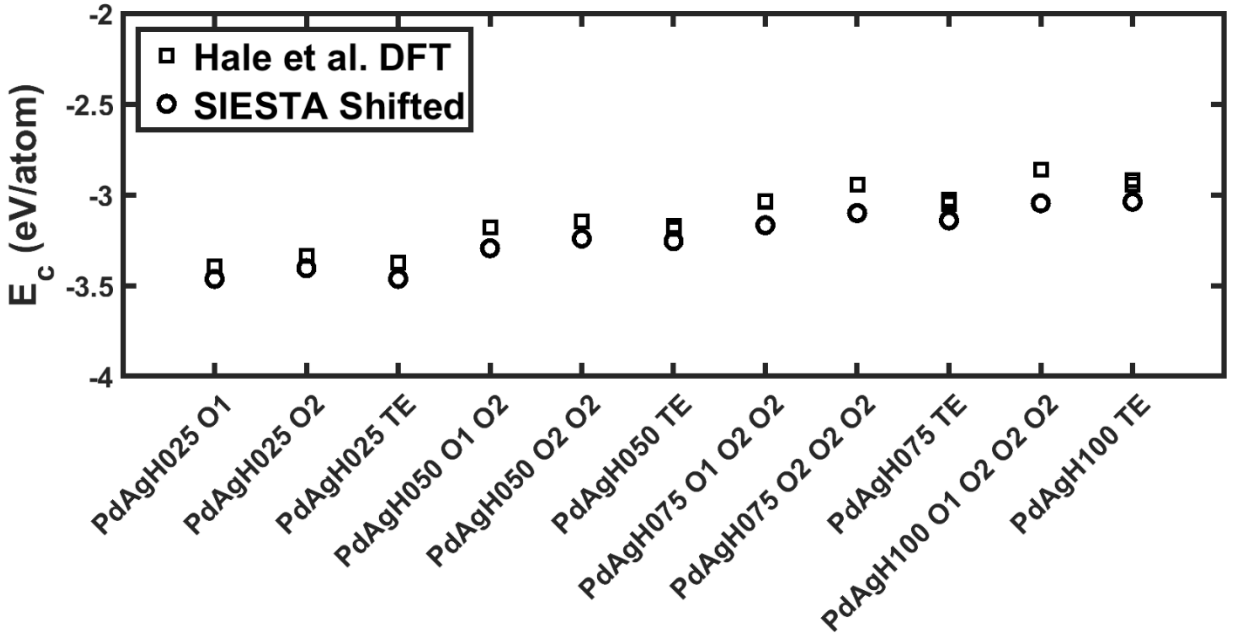

Figure 17. $Pd_{0.75}Ag_{0.25}H_x$ shifted DFT results comparison with Hale et al.

Table 10. PdAgH ab initio data, fitting results, and MD results

		SIESTA results		Shifted SIESTA		Fitting Calc	MD Results
Composition	**Structure**	**a (Å)**	**E_c (eV)**	**a (A)**	**E_c (eV)**	**E_c (eV)**	**E_c (eV)**
Pd	**FCC**	**3.9600**	**-4.9339**	**3.89**	**-3.9100**	**-3.9100**	**-3.9100**
Ag	**FCC**	**4.1443**	**-3.5480**	**4.09**	**-2.8500**	**-2.8500**	**-2.8500**
$Pd_{0.75}Ag_{0.25}$	**FCC**	**3.9981**	**-4.6392**	**3.9400**	**-3.6500**	**-3.6660**	**-3.6660**
$Pd_{0.75}Ag_{0.25}H_{0.25}$	**OC1**	**4.0451**	**-4.3728**	**4.0042**	**-3.4634**	**-3.4415**	**-3.4450**
	OC2	**4.0465**	**-4.2984**	**4.0056**	**-3.4045**	**-3.4359**	
	TE	**4.0427**	**-4.3727**	**4.0019**	**-3.4634**	**-3.3931**	
$Pd_{0.75}Ag_{0.25}H_{0.50}$	**OC1 OC2**	**4.0897**	**-4.1322**	**4.0663**	**-3.2946**	**-3.318**	**-3.3116**
	OC2 OC2	**4.1034**	**-4.0657**	**4.0800**	**-3.2416**	**-3.2423**	
	TE TE	**4.1475**	**-4.0833**	**4.1239**	**-3.2556**	**-3.2581**	
$Pd_{0.75}Ag_{0.25}H_{0.75}$	**OC1 OC2 OC2**	**4.1330**	**-3.9469**	**4.1277**	**-3.1677**	**-3.227**	**-3.2332**
	OC2 OC2 OC2	**4.1368**	**-3.8636**	**4.1315**	**-3.1008**	**-3.216**	
	TE TE TE	**4.2188**	**-3.9143**	**4.2134**	**-3.1126 to -3.1415**	**-3.0974**	
$Pd_{0.75}Ag_{0.25}H_{1.00}$	**OC1 OC2 OC2 OC2**	**4.1717**	**-3.7712**	**4.1848**	**-3.0465**	**-3.2484**	**-3.2417**
	TE TE TE TE	**4.2876**	**-3.7616**	**4.3011**	**-3.0009 to -3.0388**	**-2.8378**	

6.2 Fitting

The φ_{Ag-H} pair function takes the same mathematical form, the Morse potential, as was used successfully for the Pd-H interaction. The EAM equations for calculating the properties must be modified slightly to take into account a third atom type.

6.2.1 Cohesive Energy

The cohesive energy expression takes on an additional host term for the ternary system.

$$E_c = \frac{1}{\mathrm{x}+y+z}\left[\begin{array}{l} \mathrm{x}\left(F_{A,i}(\rho_{A,i})+\frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_A}\varphi_{A-A,ij}(r_{ij})+\frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_B}\varphi_{A-B,ij}(r_{ij})+\frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_C}\varphi_{A-C,ij}(r_{ij})\right)+ \\ y\left(F_{B,i}(\rho_{B,i})+\frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_B}\varphi_{B-B,ij}(r_{ij})+\frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_A}\varphi_{B-A,ij}(r_{ij})+\frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_C}\varphi_{B-C,ij}(r_{ij})\right)+ \\ z\left(F_{C,i}(\rho_{C,i})+\frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_C}\varphi_{C-C,ij}(r_{ij})+\frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_A}\varphi_{C-A,ij}(r_{ij})+\frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_B}\varphi_{C-B,ij}(r_{ij})\right) \end{array}\right] \tag{54}$$

$$\rho_{\mathrm{A},i} = \rho_{A-A,i}+\rho_{A-B,i}+\rho_{A-C,i} = \sum_{\substack{j=1\\ j\neq i}}^{N_A} f_A(r_{ij})+\sum_{\substack{j=1\\ j\neq i}}^{N_B} f_B(r_{ij})+\sum_{\substack{j=1\\ j\neq i}}^{N_C} f_C(r_{ij}) \tag{55}$$

$$\rho_{\mathrm{B},i} = \rho_{B-A,i}+\rho_{B-B,i}+\rho_{B-C,i} = \sum_{\substack{j=1\\ j\neq i}}^{N_A} f_A(r_{ij})+\sum_{\substack{j=1\\ j\neq i}}^{N_B} f_B(r_{ij})+\sum_{\substack{j=1\\ j\neq i}}^{N_C} f_C(r_{ij}) \tag{56}$$

$$\rho_{\mathrm{C},i} = \rho_{C-A,i}+\rho_{C-B,i}+\rho_{C-C,i} = \sum_{\substack{j=1\\ j\neq i}}^{N_A} f_A(r_{ij})+\sum_{\substack{j=1\\ j\neq i}}^{N_B} f_B(r_{ij})+\sum_{\substack{j=1\\ j\neq i}}^{N_C} f_C(r_{ij}) \tag{57}$$

Here A, B, and C represent the three types and x, y, z their respective concentrations in the structure. The expressions for the additional properties can be extended in the same manner, where an additional central atom calculation is added for the additional atom type.

6.2.2 Gibbs Free Energy of Mixing

The Gibbs free energy of mixing as a function of H concentration for PdH_x with $0 \leq x \leq 1$ was provided earlier in section 2.5. The enthalpy of mixing term can be adjusted to obtain a Gibbs free energy expression for $Pd_{1-x}Ag_xH_y$. This is the method that was employed by Hale et

al. [5]. Please note that this is not the absolute Gibbs free energy of mixing as it does not consider the heat of mixing of the $Pd_{1-x}Ag_x$ alloy or its effect on the entropy change.

$$\Delta G^{mix} = \Delta H^{mix} - \Delta S^{mix} \cdot T \tag{58}$$

$$\Delta H^{mix} = E_{Pd_{1-x}Ag_xH_y} - 2Y \cdot E_{Pd_{1-x}Ag_xH} - (1-2Y) \cdot E_{Pd_{1-x}Ag_x} \tag{59}$$

where $E_{Pd_{1-x}Ag_xH_y}$, $E_{Pd_{1-x}Ag_xH}$, $E_{Pd_{1-x}Ag_x}$, are the cohesive energies, and Y = y/(1+y) is the mole fraction.

$$\Delta S_t = -k_B \cdot \begin{bmatrix} Y \cdot \ln[Y/(1-Y)] + \\ (1-2 \cdot Y) \cdot \ln[(1-2Y)/(1-Y)] \end{bmatrix} \tag{60}$$

where k_B is Boltzmann's constant.

6.2.3 Fitting Results

Fitting is performed with constrained nonlinear optimization in MATLAB. Validation of the code is discussed in Appendix C. The code is given in Appendix D. The fitting parameters obtained for the Ag-H interatomic potential are listed in table 9, and the pair potential functions used in the model are plotted in figures 18 and 19. The cohesive energy from fitting is in good agreement with the fitting data for most of the structures; however at the highest concentrations the quality of the fit diminishes.

Table 11. Fitting parameters for Ag-H

D_{AgH}	α_{AgH}	β_{AgH}	$r_{0,AgH}$
1.476745	**1.967649**	**1.741865**	**1.850017**

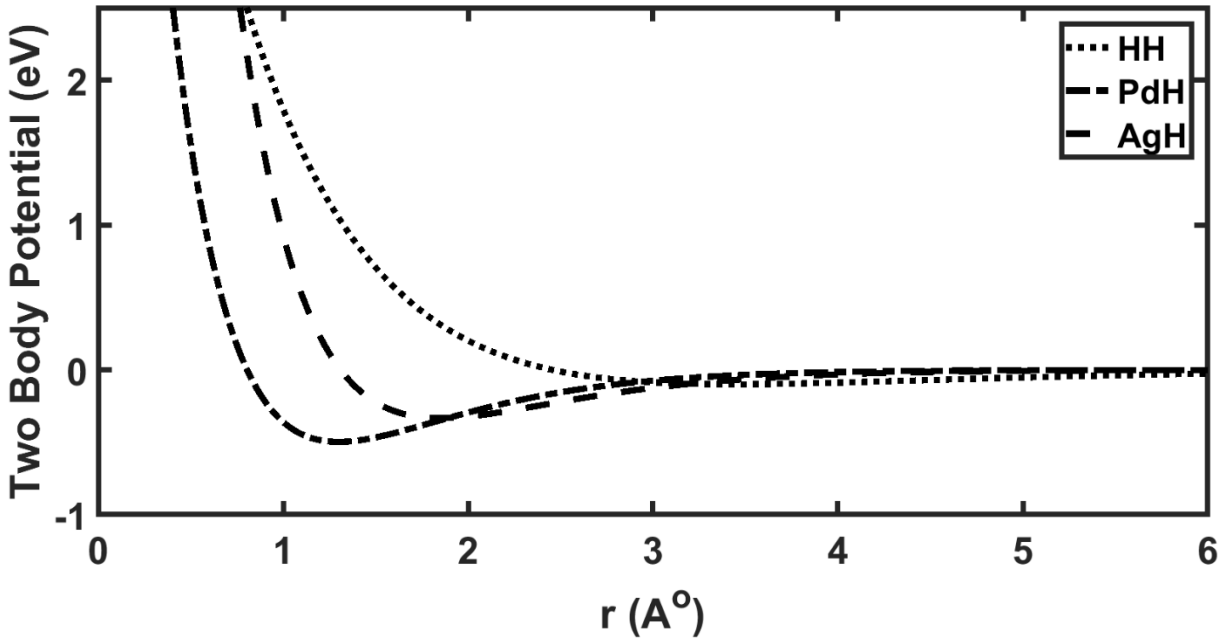

Figure 18. Pair potential functions of the fitted H-H, Pd-H, and Ag-H potentials.

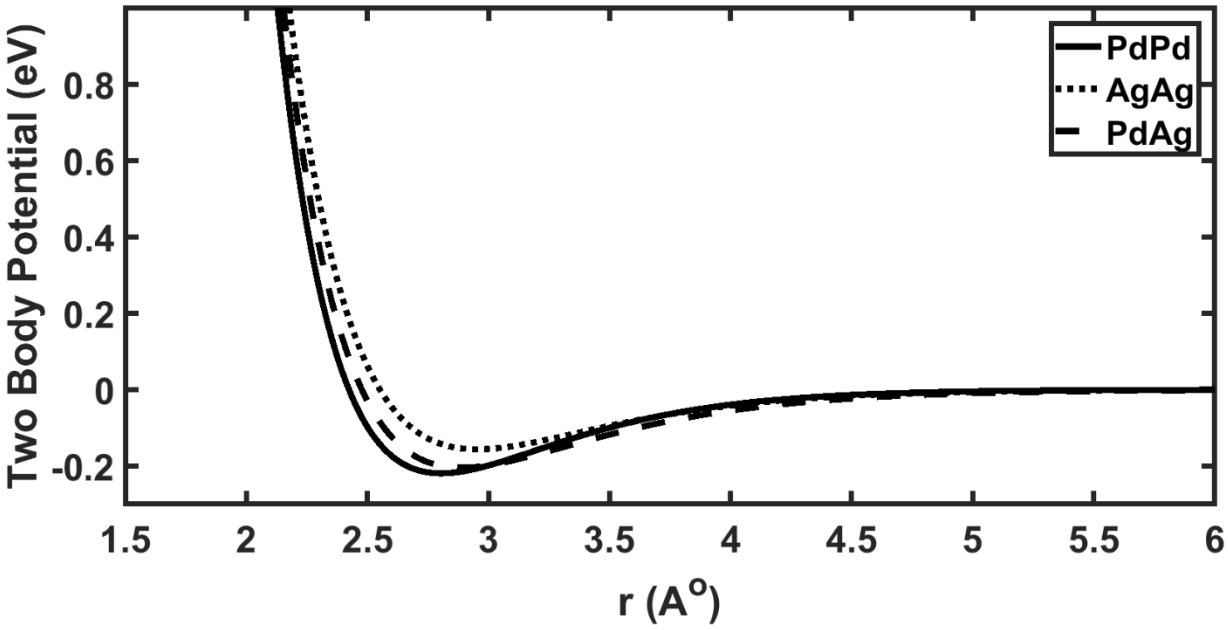

Figure 19. Pair potential functions of the fitted Pd-Pd, Ag-Ag, and Pd-Ag potentials.

6.3 Results

The fitted potentials were used to generate a tabulated potential file in DYNAMO *setfl* format utilizing a MATLAB code written for this purpose (see Appendix D). Numerous MD simulations were performed for the PdAgH structures using the generated potential file with LAMMPS.

6.3.1 Lattice Constants and Cohesive Energy

The lattice constants obtained for $Pd_{0.75}Ag_{0.25}H_x$ were in good agreement with the results of DFT used in the fitting procedure, as seen in figure 20. The cohesive energies, plotted in figure 21, were in good agreement with the results from the MATLAB fitting calculations, and in good agreement with the DFT data at lower hydrogen concentrations. For the higher hydrogen compositions, the cohesive energies begin to deviate from the DFT results, however, the

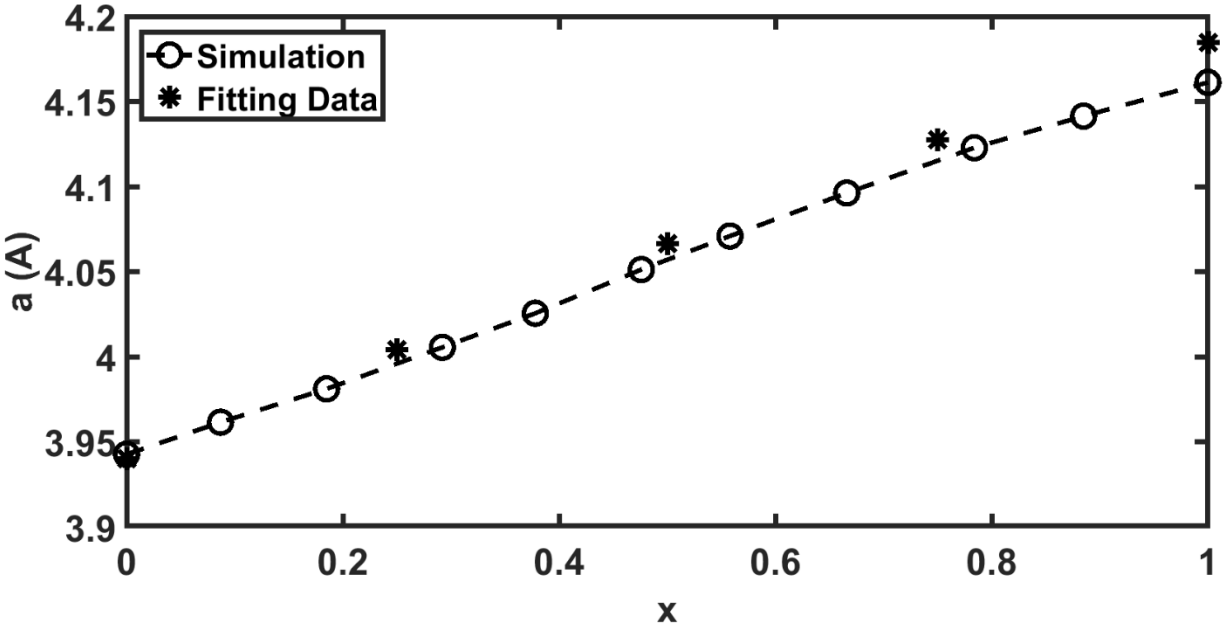

Figure 20. $Pd_{0.75}Ag_{0.25}H_x$ lattice constants from MD and Fitting Data.

consistency between the fitting calculations and MD suggest that improving the quality of the fit would reduce this deviation. Ongoing work will focus on improving the quality of the fitting to make the potentials more reliable at higher H compositions.

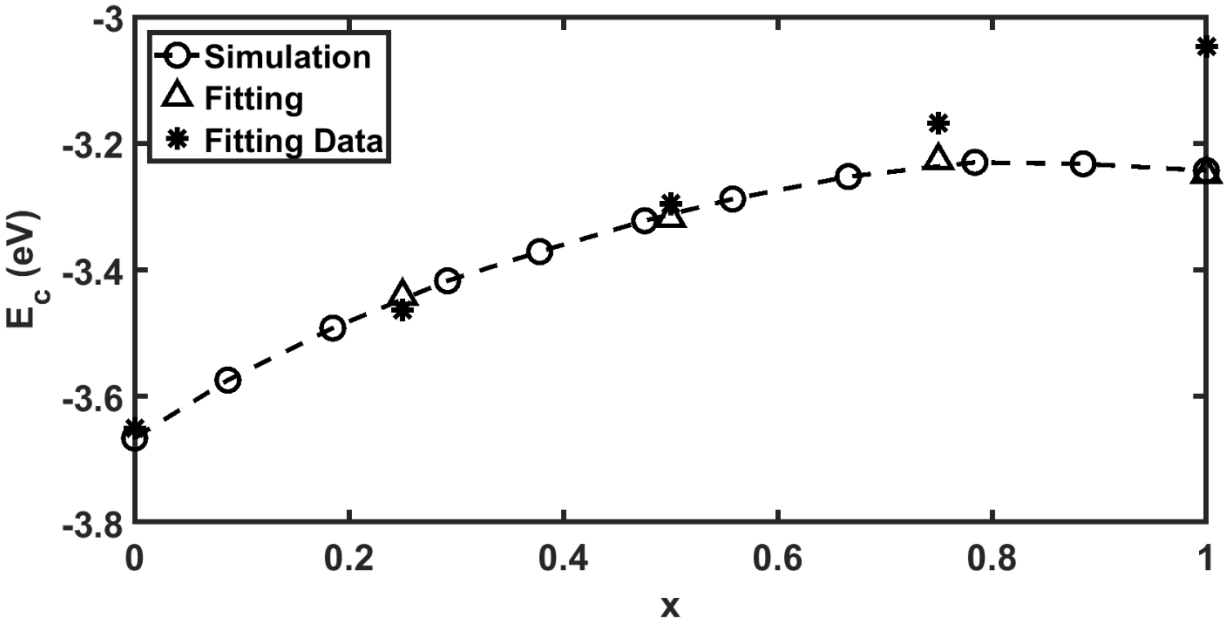

Figure 21. $Pd_{0.75}Ag_{0.25}H_x$ cohesive energies from MD, Fitting, and Fitting Data.

6.3.2 Elastic Constants and Bulk Modulus

The bulk modulus from MD simulation is shown in figure 22, the expected softening trend with H concentration is observed [38,24,44]. This result is also in good agreement with those obtained by Hale et al. [5] for PdAgH alloys. In figure 23 we see that the shear elastic constant for $Pd_{0.75}Ag_{0.25}H_x$ increases with H composition, matching the behavior obtained experimentally for PdH by Schwarz et al. [38]. This is reasonable behavior for our structures due to the large percentage of Pd present. Figures 24 and 25 show that the C_{11} and C_{12} elastic constants for both $Pd_{1.00}$ H_x and $Pd_{0.75}Ag_{0.25}H_x$ display smooth curves with similar trends. These

results demonstrate the reliability of the potentials for predicting the mechanical properties of the alloys.

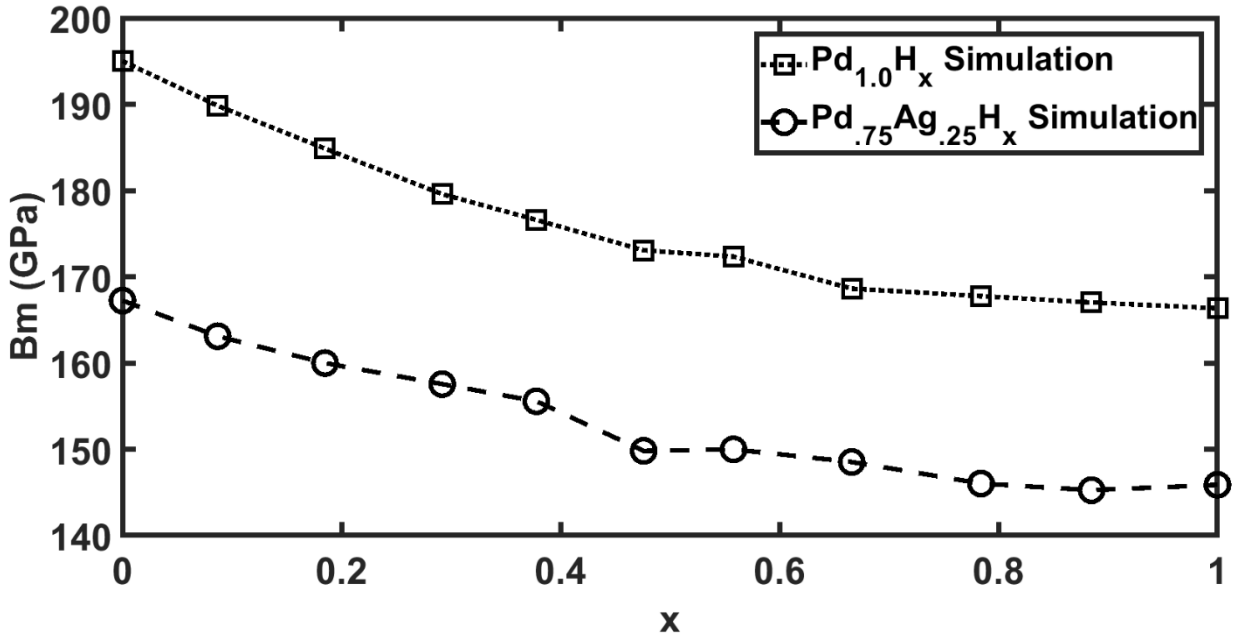

Figure 22. $Pd_{1.00}H_x$ and $Pd_{0.75}Ag_{0.25}H_x$ bulk modulus from MD.

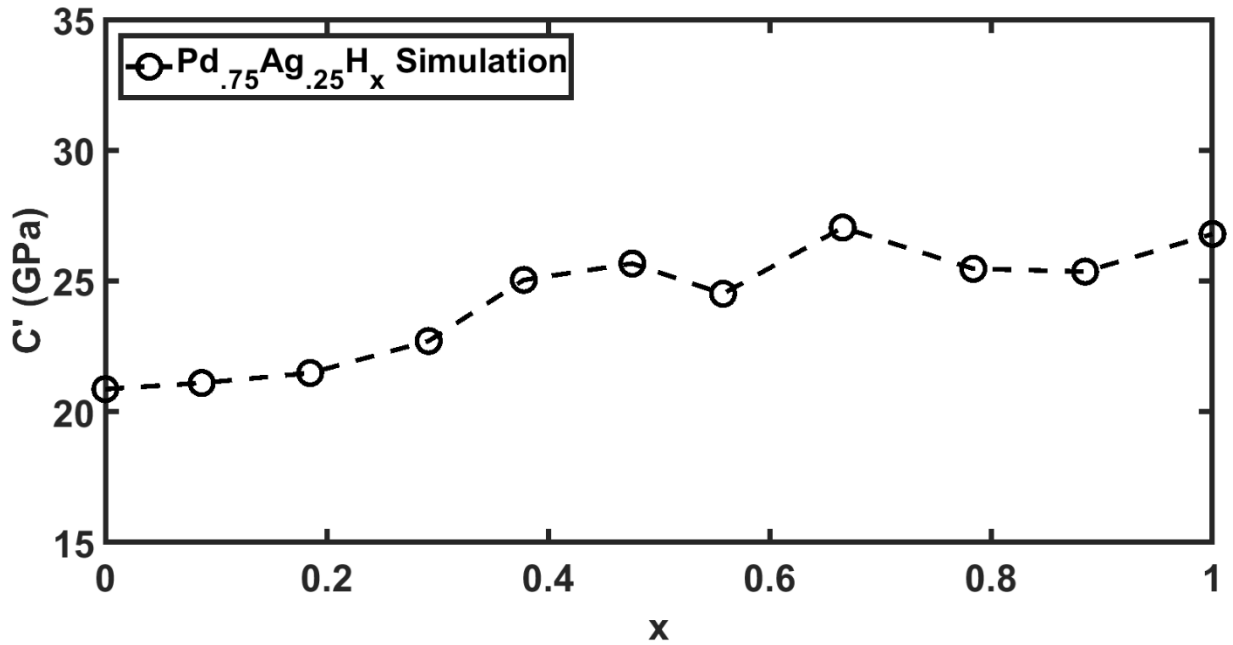

Figure 23. $Pd_{0.75}Ag_{0.25}H_x$ C' shear elastic constant from MD.

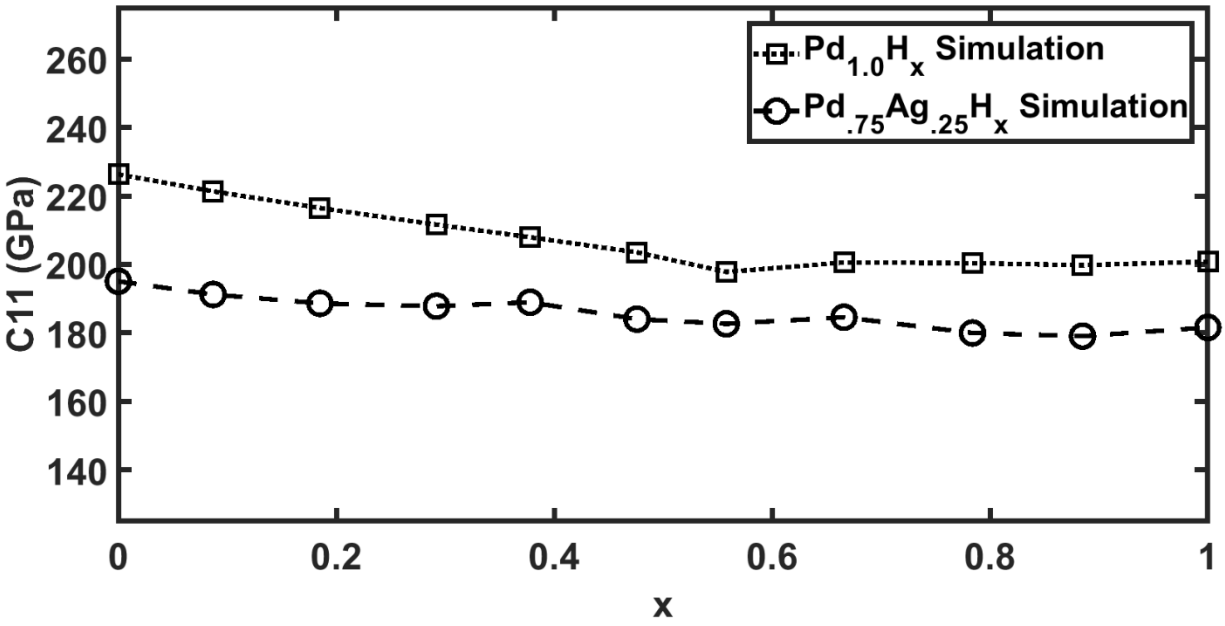

Figure 24. $Pd_{1.00}H_x$ and $Pd_{0.75}Ag_{0.25}H_x$ C_{11} elastic constant from MD.

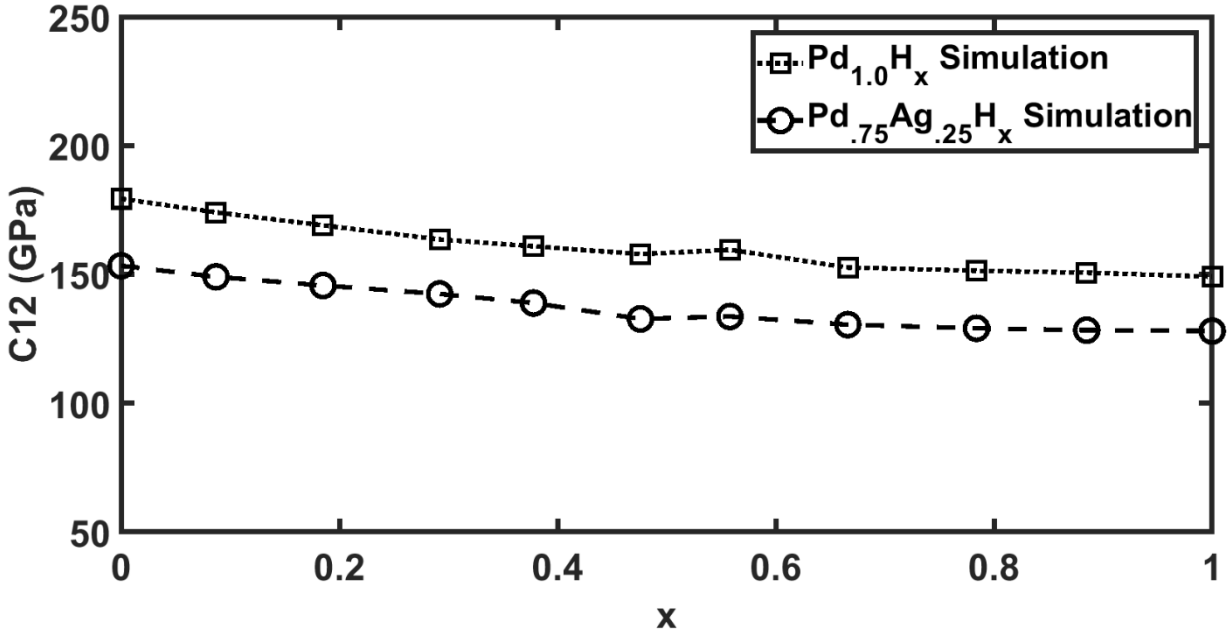

Figure 25. $Pd_{1.00}H_x$ and $Pd_{0.75}Ag_{0.25}H_x$ C_{12} elastic constant from MD.

6.3.3 Additional Compositions

Figures 26 and 27 show the lattice constant and cohesive energy for several different alloys over the full hydrogen composition range. This illustrates the versatility of this EAM model even outside of the silver concentrations used in fitting. The lattice constant can be seen to show a consistent trend and good agreement with the fitting data. Similarly, the cohesive energy data is smooth and shows a consistent trend as the Ag and H concentrations are increased. The agreement with the fitting data is very good for most of the hydrogen composition range. The range $0.5 < x < 1.0$ may be improved with further refinement of the fitting.

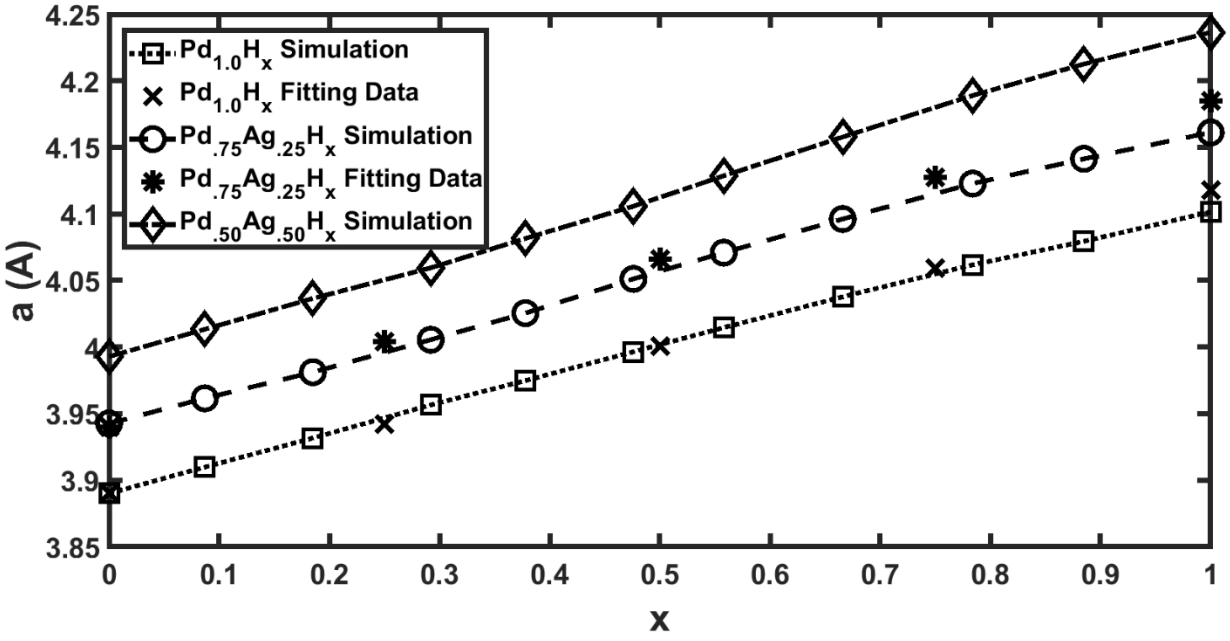

Figure 26. $Pd_{1.00}H_x$, $Pd_{0.75}Ag_{0.25}H_x$, and $Pd_{0.50}Ag_{0.50}H_x$ lattice constants from MD.

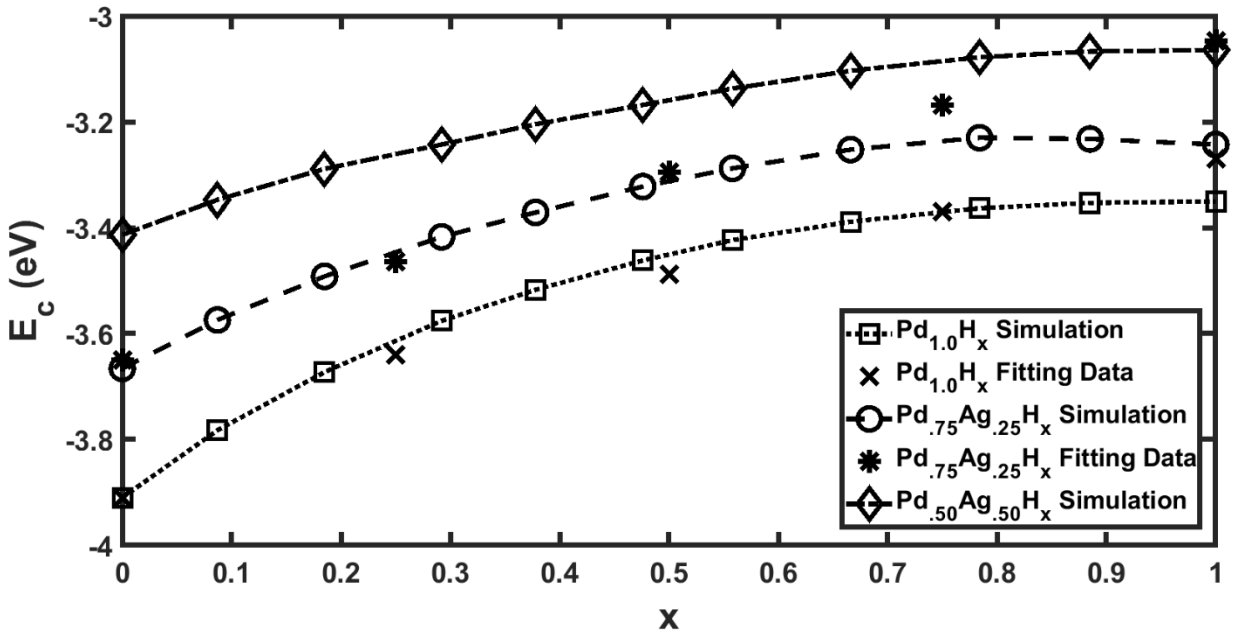

Figure 27. $Pd_{1.00}H_x$, $Pd_{0.75}Ag_{0.25}H_x$, and $Pd_{0.50}Ag_{0.50}H_x$ cohesive energies from MD.

6.3.4 Dynamic Stability

In palladium hydride, H atoms prefer to occupy OC interstitial positions in the Pd fcc structure [40]. The DFT results indicate that the OC positions are likewise energetically favorable for hydrogen in $Pd_{.75}Ag_{.25}H_x$; this is consistent with the findings of Hale et al. and earlier by Løvvik and Olsen [5,26]. To test the structural stability for $Pd_{1-x}Ag_xH_y$ structures with our EAM model, LAMMPS was used to create various structures with H atoms occupying TE positions as shown in figure 28(a). Using MD simulation with an NPT ensemble each TE structure was annealed for 100 ns from 500 K to 1 K, this was followed by CG energy minimization. At the end of each MD + MS simulation, the hydrogen atoms moved to the lower energy OC interstitial positions as predicted. The resulting structure is shown in figure 28(b).

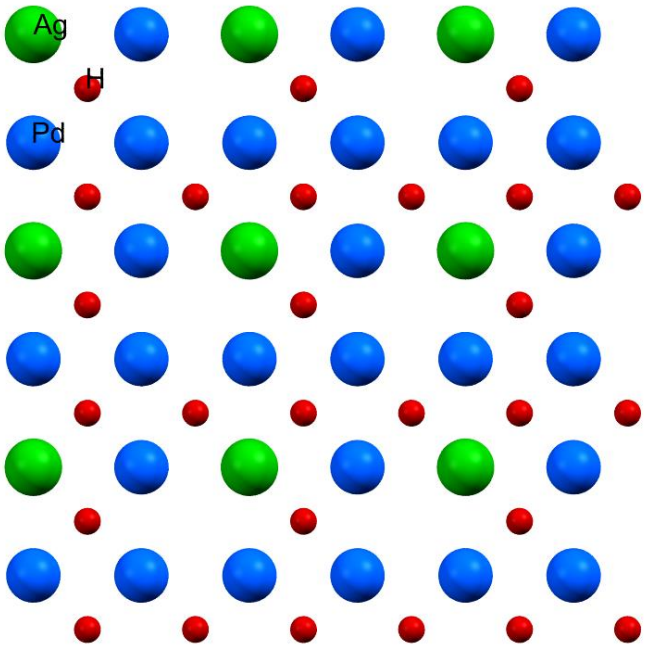

Figure 28. (a) TE structure before simulation. Interstitial H (red) within the Pd (blue) and Ag (green) lattice.

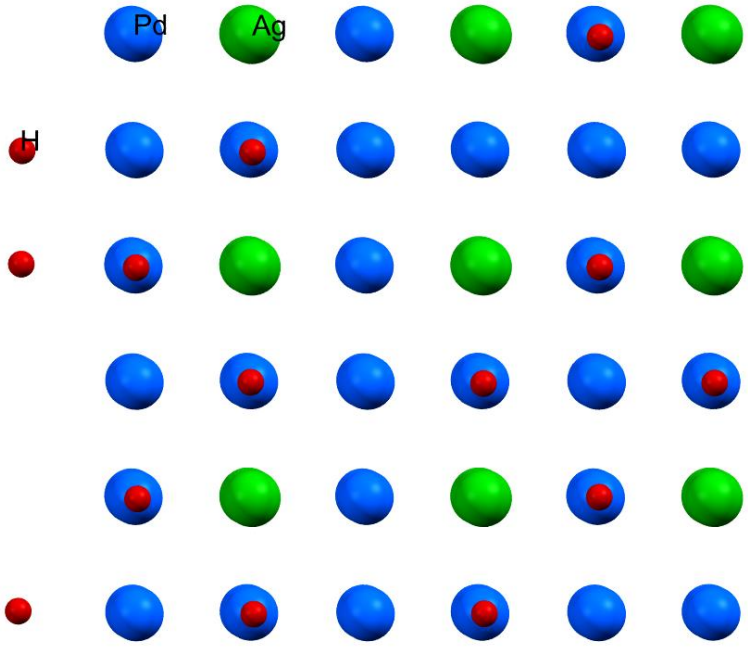

Figure 28. (b) OC structure after simulation. Interstitial H (red) within the Pd (blue) and Ag (green) lattice.

6.3.5 Miscibility Gap

Figure 29 shows that, as expected at 0 K, the Gibbs free energy is positive for all alloys and hydrogen compositions, implying that the attractive interactions between atoms of different species are, on average, weaker than those between molecules of the same species. At room temperature, experimental results indicate that the miscibility gap decreases as the silver concentration increases. The α and β phases cease to be distinct around $Pd_{1-x}Ag_x$, x = 0.25–0.30 [18-20]. The curves shown in figure 30 show that as the silver concentration increases, the miscibility gap narrows and the values become more negative indicating more favorable mixing. At x = 0.5 no miscibility gap is found. This shows the model is capable of predicting the miscibility gap and the behavior of the model with increasing silver concentration is consistent with the trend of the experimental data.

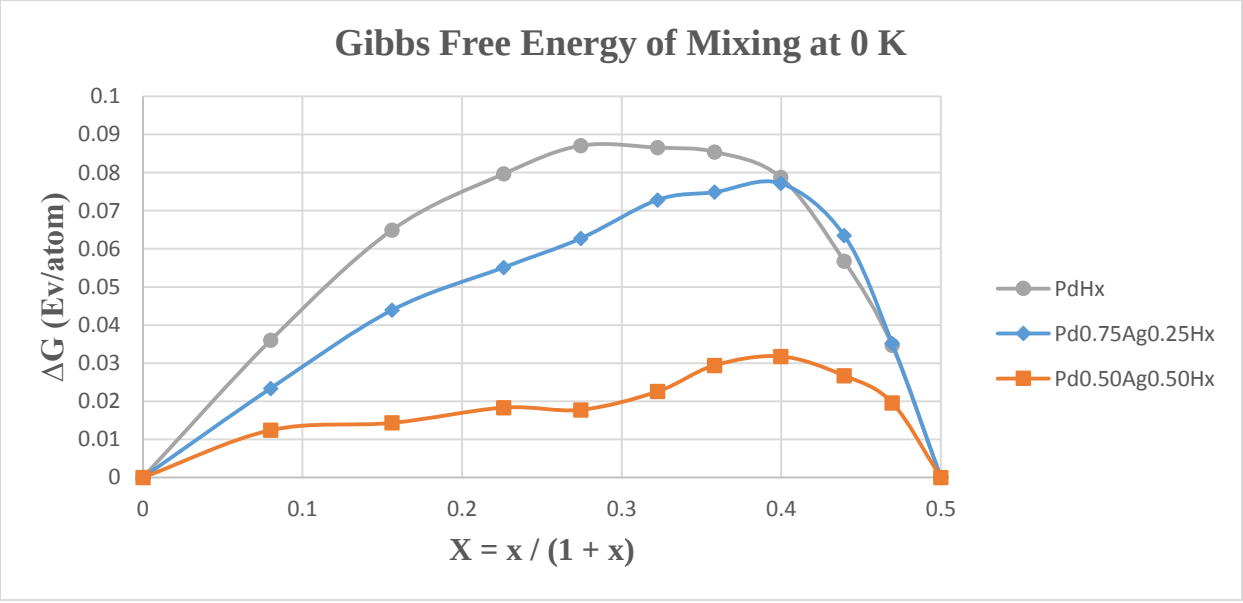

Figure 29. Free energy of mixing associated with addition of hydrogen at 0 K.

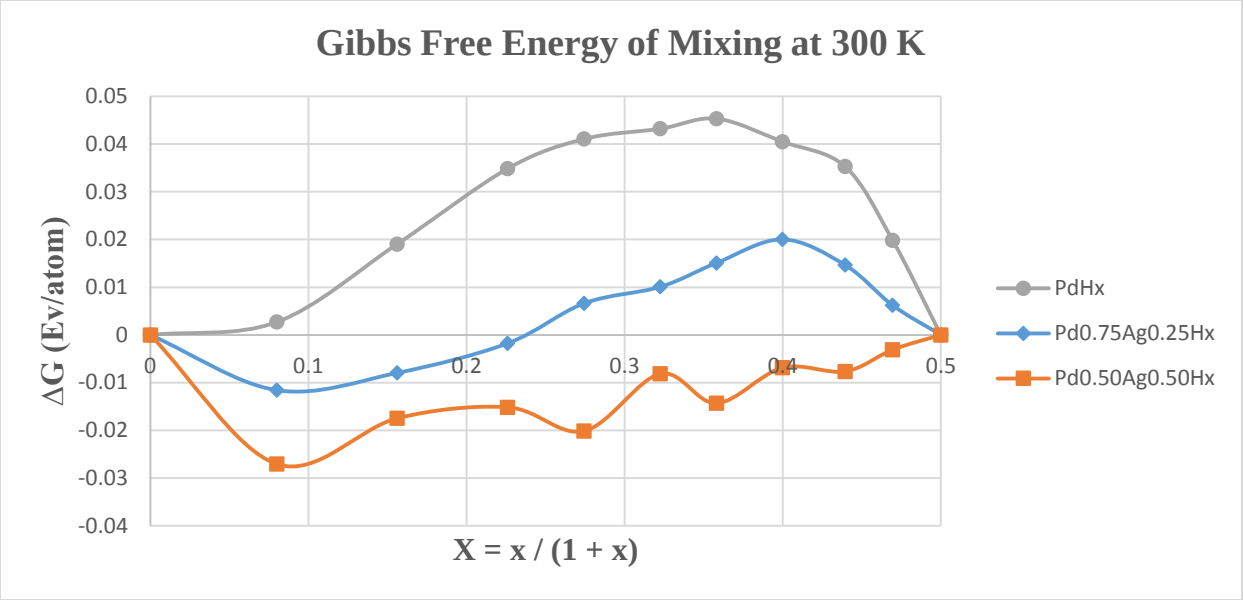

Figure 30. Free energy of mixing associated with addition of hydrogen at 300 K.

CHAPTER 7

CONCLUSION

In this work, a fully analytical interatomic EAM potential for the palladium-silver hydride system was developed. The proposed potential has fewer fitting parameters than previously developed EAM Pd-Ag-H potentials [5] and behaves well in minimizing the objective function during the fitting procedures. The Pd-Pd and Ag-Ag potentials each have six fitting parameters. The mixing rule defines the Pd-Ag interaction and has two fitting parameters. The validation results indicated that each of these potentials can be used reliably in MD simulations. The H-H potential has 10 fitting parameters, and the Pd-H and Ag-H potentials each have four.

The Pd-Ag potential was fitted to experimental data for the heat of solution. The Ag-H potential was fitted to the cohesive energies of 14 $Pd_{0.75}Ag_{0.25}H_y$ structures obtained from ab initio calculations with SIESTA. MD testing with LAMMPS shows that the lattice constants and cohesive energies for PdAgH are in good agreement with the results from ab initio simulations for much of the hydrogen composition range. The elastic properties also show the expected behavior, with the bulk modulus decreasing with hydrogen composition. Dynamic stability testing shows hydrogen atoms moving from TE positions to lower energy OC positions as expected. The model also captures the miscibility gap and shows that it narrows and disappears with increasing silver concentration as predicted.

This work shows that an EAM model can be fit for the PdAgH system using the central atom method without the need for much more time consuming molecular dynamics simulations in the fitting procedure. In conclusion, this EAM interatomic potential reliably captures many properties of the Pd-Ag-H systems while being less programmatically complex and less computationally intensive than earlier models.

REFERENCES

[1] Nelin, G. "A neutron diffraction study of palladium hydride." *Phys Status Solidi B* (1971).

[2] Caputo, R, and Alavi, A. "Where do the H atoms reside in PdH_x systems." *Mol Phys* (2003).

[3] Manchester, F D, San-Martin, A, and Pitre, J M. "The H-Pd (hydrogen palladium) system." *J Phase Equilib* (1994).

[4] Zhou, X W, Zimmerman, J A, Wong, B M, and Hoyt, J J. "An embedded-atom method interatomic potential for Pd-H alloy." *J. Mater.* vol. 23, (2008): pp. 704-718.

[5] Hale, L M, Wong, B M, Zimmerman, J A, and Zhou, X W. "Atomistic potentials for palladium–silver hydrides." *Modelling and Simulation in Materials Science and Engineering* vol. 21, no. 4 (2013).

[6] Hsu, D K, Leisure, R G. "Elastic constants of palladium and β phase palladium hydride between 4 and 300 K." *Phys.Rev. B.* vol. 20 (1979): pp**.** 1339.

[7] Dong, W, Ledentu, V, Sautet, P, Eichler, A, and Hafner, J. "Hydrogen adsorption on palladium: a comparative theoretical study of different surfaces." *Surf Sci.* (1998).

[8] Goltsova, M V, Artemenko, Y. A., and Zaitsev, V. I. "Kinetics and morphology of the reverse β/α hydride transformation in thermodynamically open Pd-H system." *J Alloys Compd* (1999).

[9] Lasser, R. *Tritium and Helium-3 in Metals*. Vol. 9, Springer-Verlag, Berlin (1989).

[10] Alefeld, G and Völkl, J. *Hydrogen in Metals*. Vol. 1, Springer-Verlag, Berlin, (1978).

[11] Alefeld, G and Völkl, J. *Hydrogen in Metals*. Vol. 2, Springer-Verlag, Berlin, (1978).

[12] Muetterties, E.L. *Transition Metal Hydrides*. Marcel Dekker, New York (1971).

[13] Siegel, B, Libowitz, G G, Mueller, W M, and Blackledge, J P. *Metal Hydrides.* Academic Press, New York (1968).

[14] Fukai, Y. *The Metal-Hydrogen System.* 21, Springer-Verlag, Berlin (1993).

[15] Povel, R, Feucht, K, Gelse, W, and Withalm, G. *Hydrogen fuel for motorcars*. *Interdiscip. Sci.Rev.* vol. 14, no. 4 (1989): pp. 365.

[16] Knapton, A G, "Palladium alloys for hydrogen diffusion membranes." *Platinum Met.* 21, (1977): pp. 44–50.

[17] Wolf, R J, Lee, M W, Davis, R C, Fay, P J, and Ray, J R. "Pressure–composition isotherms for palladium hydride." *Phys. Rev. B.* vol. 48, (1993): pp. 12415.

[18] Wise, M L H, Farr, J P G and Harris, I R. "X-ray studies of alpha-beta miscibility gaps of some palladium solid solution hydrogen systems." *J. Less-Common Met.* vol. 41, (1975): pp. 115–27.

[19] Axelrod, S D, and Makrides, A C. "X-ray studies of hydrogen–silver–palladium electrodes." *J. Phys. Chem.* vol. 68, (1964): pp. 2154–9.

[20] Burch, R. "On role of silver atoms in absorption of hydrogen by palladium–silver alloys." *Solid State Commun.* vol. 7, (1969): pp. 1313–7.

[21] Foletto, E L, Wirbitzki Da Silveira, J V, and Jahn, S L. "Preparation of palladium–silver alloy membranes for hydrogen permeation." *Latin Am. Appl. Res.* vol. 38, (2008): pp. 79–84.

[22] Uemiya, S, Matsuda, T, and Kikuchi, E. "Hydrogen permeable palladium silver alloy membrane supported on porous ceramics" *J. Membr. Sci.* vol. 56, (1991): pp. 315–25.

[23] Holleck, G L. "Diffusion and solubility of hydrogen in palladium and palladium–silver alloys." *J. Phys. Chem.* vol. 74, (1970): pp. 503–11.

[24] Ilawe, V, Niranjan, Zimmerman, A J, and Wong, B M. "Breaking Badly: DFT-D2 Gives Sizeable Errors for Tensile Strengths in Palladium-Hydride Solids." *Journal of chemical theory and computation.* (2015): pp. 5426−5435.

[25] Wei, Cai, Kong, F T, and Gong, H R "Phase stability and elastic property of PdH and PdCuH phases." *International Journal of Hydrogen Energy.* vol. 38, no. 36 (2013): pp. 16485-16494.

[26] Løvvik, O M, and Olsen, R A. "Density functional calculations on hydrogen in palladium–silver alloys." *Journal of alloys and compounds.* vol 330 (2002): pp. 332-337.

[27] Sonwane, Chandrashekhar G., Jennifer Wilcox, and Yi Hua Ma. "Solubility of hydrogen in PdAg and PdAu binary alloys using density functional theory." *The Journal of Physical Chemistry B.* vol. 110, no. 48 (2006): pp. 24549-24558.

[28] Foiles, S M, and Hoyt, J J. "Computer Simulation of Bubble Growth in Metals Due to He." *Sandia National Laboratories*. (2001).

[29] Hijazi, I, Zhang, Y, Fuller, R. "A Simple EAM potential for Hydrogen-Selective Palladium Based Membranes for Biomass Derived Syngas Processing." *Proceedings of the ASME 2018 12th International Conference on Energy Sustainability*. ES2018-7369.

[30] Daw, M S, and Baskes, M I. "Semiempirical, quantum mechanical calculation of hydrogen embrittlement in metals." *Phys. Rev. Lett.* vol. 50, (1983): pp. 1285.

[31] Hijazi, I A, and Park, Y H. "Consistent analytic embedded atom potential for face-centered cubic metals and alloys." *J. Matter. Sci. Technol.* vol. 25, (2009): pp. 835-846.

[32] Park, Y H, and Hijazi, I A. "Simple Analytic Embedded Atom Potential for FCC Materials." *International Journal of Microstructure and Materials Properties.* vol. 6, no. 5 (2011).

[33] Hijazi, I A, and Park, Y H. "Structure of Pure Metallic Nanoclusters: Monte Carlo Simulation and Ab Initio Study." *The European Physical Journal D*. vol. 59, (2010): pp. 215.

[34] Hijazi, I A, and Park, Y H. "Critical Size of Transitional Copper Clusters for Ground State Structure Determination: Empirical and Ab Initio Study." *Molecular Simulation*. vol. 38, no. 3, (2012): pp. 241.

[35] Hijazi, I A, and Park, Y H. "Structural, Electronic and Magnetic Properties of Pure Metallic and Bimetallic Nanoclusters: Empirical and Density Functional Studies." *Molecular Simulation.* 38 no. 6, (2013): pp. 505.

[36] Rose, J H, Smith, J R, Guinea, F, and Ferrante, J. "Universal features of the equation of state of metals." *Phys. Rev. B.* vol. 29 (6), (1984): pp. 2963.

[37] Puska, J, Nieminen, R M, and Manninen, M. "Atoms embedded in an electron gas: immersion energies." *Physical Review B.* vol. 24, no. 6, (1981): pp. 3037–3047.

[38] Schwarz, R B, Bach, H T, Harms, U, and Tuggle, D. "Elastic properties of Pd–hydrogen, Pd–deuterium, and Pd–tritium single crystals." *Acta Mater.* vol. 53, (2005): pp. 569-580.

[39] Sakamoto,Y, Yuwasa, K, and Hirayama, K. "X-ray investigation of the absorption of hydrogen by several palladium and nickel solid solution alloys." *J. Less-Comm. Metals*. vol. 88, (1982): pp. 115.

[40] Antonov, V E. "Phase transformations, crystal and magnetic structures of high-pressure hydrides of d-metals." *Journal of alloys and compounds.* vol. 330, (2002): pp. 110-116.

[41] Jacobs, M H, and Oonk, H A. "The calculation of ternary miscibility gaps using the linear contributions method: Problems, benchmark systems and an application to (K, Li, Na) Br." *Calphad*. vol. 30, no. 2 (2006): pp. 185-190.

[42] Goods, S H, and Guthrie, S E. "Mechanical properties of palladium and palladium hydride." *Scripta Metall. Mater.* vol. 26**,** no. 561 (1992).

[43] Thompson, A. "in.elastic." Sandia National Laboratory. Available from: https://github.com/lammps/lammps/tree/master/examples/ELASTIC

[44] Zhong, W, Cai, Y, and Tomanek, D. "Computer simulation of hydrogen embrittlement in metals." *Nature.* vol. 362, no. 6419 (1993): pp. 435-437.

[45] Johnson, R A. "Alloy models with the embedded-atom method." *Physical Review B.* vol. 39, no. 17 (1989): pp. 12554.

[46] Foiles, S M, Baskes, M I, and Daw, M S. "Embedded-atom-method functions for the fcc metals Cu, Ag, Au, Ni, Pd, Pt, and their alloys." *Physical review B.* 33, no. 12 (1986): 7983.

[47] Delczeg-Czirjak, EK, Delczeg, L, Ropo, M, Kokko, K, Punkkinen, MP, Johansson, B, and Vitos, L. "Ab initio study of the elastic anomalies in Pd-Ag alloys." *Physical Review B.* vol. 79, no. 8 (2009): pp. 085107.

[48] Karakaya, I, and Thompson, W T. "The Ag−Pd (Silver-Palladium) system." *Bulletin of alloy phase diagrams.* vol. 9, no. 3 (1988): pp. 237-243.

[49] Zhou, X W, Zimmerman, J A, Hoyt, J J, Leonard, F L, and Griffin, J D. *Development of an inter-atomic potential for the Pd-H binary system* (No. SAND2007-5976). Sandia National Laboratories. (2007).

APPENDIX A:

INSTITUTIONAL REVIEW BOARD LETTER

MARSHALL UNIVERSITY
www.marshall.edu

Office of Research Integrity

May 23, 2018

Robert Fuller
1628 7th Ave., Apt 12
Huntington, WV 25703

Dear Mr. Fuller:

This letter is in response to the submitted thesis abstract entitled *"Improved Embedded Atom Potentials for Metal Hydride Systems."* After assessing the abstract, it has been deemed not to be human subject research and therefore exempt from oversight of the Marshall University Institutional Review Board (IRB). The Code of Federal Regulations (45CFR46) has set forth the criteria utilized in making this determination. Since the information in this study does not involve human subjects as defined in the above referenced instruction, it is not considered human subject research. If there are any changes to the abstract you provided then you would need to resubmit that information to the Office of Research Integrity for review and a determination.

I appreciate your willingness to submit the abstract for determination. Please feel free to contact the Office of Research Integrity if you have any questions regarding future protocols that may require IRB review.

Sincerely,

Bruce F. Day, ThD, CIP
Director

APPENDIX B:

CALCULATIONS FOR INTERSTITIAL SOLID SOLUTION

$$r_{ij} = \sqrt{r_i^2 + r_j^2 + r_k^2} = \sqrt{l_1^2 + l_2^2 + l_3^2}\, a = M_{ij} a \tag{1}$$

$$\frac{\partial r_{ij}}{\partial a} = M_{ij} = \frac{r_{ij}}{a} \tag{2}$$

$$\frac{\partial^2 r_{ij}}{\partial a^2} = 0 \tag{3}$$

$$\frac{\partial r_{ij}}{\partial r_i} = \frac{r_i}{\sqrt{r_i^2 + r_j^2 + r_k^2}} = \frac{r_i}{r_{ij}} \tag{4}$$

$$\frac{\partial^2 r_{ij}}{\partial r_i^2} = \frac{r_{ij} - r_i^2 / r_{ij}}{r_{ij}^2} = \frac{1}{r_{ij}} - \frac{r_i^2}{r_{ij}^3} \tag{5}$$

$$\frac{\partial^2 r_{ij}}{\partial r_i \partial r_j} = \frac{0 - r_i r_j / r_{ij}}{r_{ij}^2} = -\frac{r_i r_j}{r_{ij}^3} \tag{6}$$

$$\frac{\partial^2 r_{ij}}{\partial r_i \partial r_j} = \delta_{ij} \frac{1}{r_{ij}} - \frac{r_i r_j}{r_{ij}^3}, \quad \delta_{ij} = \begin{cases} 1 & , i = j \\ 0 & , i \neq j \end{cases} \tag{7}$$

$$\frac{\partial E_c}{\partial a} = \frac{\partial E_c}{\partial r_{ij}} \frac{\partial r_{ij}}{\partial a} = \frac{\partial E_c}{\partial r_{ij}} \frac{r_{ij}}{a} \tag{8}$$

$$\frac{\partial^2 E_c}{\partial a^2} = \frac{\partial}{\partial a}\left(\frac{\partial E}{\partial r_{ij}} \frac{\partial r_{ij}}{\partial a}\right) = \frac{\partial}{\partial a}\left(\frac{\partial E}{\partial r_{ij}}\right)\frac{\partial r_{ij}}{\partial a} + \frac{\partial}{\partial a}\left(\frac{\partial r_{ij}}{\partial a}\right)\frac{\partial E}{\partial r_{ij}}$$

$$= \frac{\partial}{\partial r_{ij}} \frac{\partial r_{ij}}{\partial a}\left(\frac{\partial E}{\partial r_{ij}}\right)\frac{\partial r_{ij}}{\partial a} + \frac{\partial r_{ij}^2}{\partial a^2} \frac{\partial E}{\partial r_{ij}}$$

$$= \frac{\partial^2 E_c}{\partial r_{ij}^2}\left(\frac{\partial r_{ij}}{\partial a}\right)^2 + 0 \cdot \frac{\partial E}{\partial r_{ij}}$$

$$= \frac{\partial^2 E_c}{\partial r_{ij}^2}\left(\frac{\partial r_{ij}}{\partial a}\right)^2$$

$$= \frac{\partial^2 E_c}{\partial r_{ij}^2}\left(\frac{r_{ij}}{a}\right)^2 \quad (9)$$

$$\rho_{H,i} = \rho_{H-H,i} + \rho_{H-I,i} \quad (10a)$$

$$\rho_{I,i} = \rho_{I-H,i} + \rho_{I-I,i} \quad (10b)$$

$$\frac{\partial \rho_{H,i}}{\partial r_{ij}} = \frac{\partial \rho_{H\text{-}H,i}}{\partial r_{ij}} + \frac{\partial \rho_{H-I,i}}{\partial r_{ij}} \quad (11a)$$

$$\frac{\partial \rho_{I,i}}{\partial r_{ij}} = \frac{\partial \rho_{I\text{-}I,i}}{\partial r_{ij}} + \frac{\partial \rho_{I-H,i}}{\partial r_{ij}} \quad (11b)$$

$$\frac{\partial^2 \rho_{H,i}}{\partial r_{ij}^2} = \frac{\partial^2 \rho_{H-H,i}}{\partial r_{ij}^2} + \frac{\partial^2 \rho_{H-I,i}}{\partial r_{ij}^2} \quad (12a)$$

$$\frac{\partial^2 \rho_{I,i}}{\partial r_{ij}^2} = \frac{\partial^2 \rho_{I-I,i}}{\partial r_{ij}^2} + \frac{\partial^2 \rho_{I-H,i}}{\partial r_{ij}^2} \quad (12b)$$

$$\phi_{H,i} = \phi_{H-H,i} + \phi_{H-I,i} \quad (13a)$$

$$\phi_{I,i} = \phi_{I-H,i} + \phi_{I-I,i} \quad (13b)$$

$$\frac{\partial \phi_{H,i}}{\partial r_{ij}} = \frac{\partial \phi_{H-H,i}}{\partial r_{ij}} + \frac{\partial \phi_{H-I,i}}{\partial r_{ij}} \quad (14a)$$

$$\frac{\partial \phi_{I,i}}{\partial r_{ij}} = \frac{\partial \phi_{I-I,i}}{\partial r_{ij}} + \frac{\partial \phi_{I-H,i}}{\partial r_{ij}} \quad (14b)$$

$$\frac{\partial^2 \phi_{H,i}}{\partial r_{ij}^2} = \frac{\partial^2 \phi_{H-H,i}}{\partial r_{ij}^2} + \frac{\partial^2 \phi_{H-I,i}}{\partial r_{ij}^2} \quad (15a)$$

$$\frac{\partial^2 \phi_{I,i}}{\partial r_{ij}^{\ 2}} = \frac{\partial^2 \phi_{I-I,i}}{\partial r_{ij}^{\ 2}} + \frac{\partial^2 \phi_{I-H,i}}{\partial r_{ij}^{\ 2}} \tag{15b}$$

$$\frac{\partial F_H}{\partial r_{ij}} = \frac{\partial F_H}{\partial \rho_{H,i}} \frac{\partial \rho_{H,i}}{\partial r_{ij}} \tag{16a}$$

$$\frac{\partial F_I}{\partial r_{ij}} = \frac{\partial F_I}{\partial \rho_{I,i}} \frac{\partial \rho_{I,i}}{\partial r_{ij}} \tag{16b}$$

$$\frac{\partial^2 F_H}{\partial r_{ij}^{\ 2}} = \frac{\partial}{\partial r_{ij}}\left(\frac{\partial F_H}{\partial \rho_{H,i}} \frac{\partial \rho_{H,i}}{\partial r_{ij}}\right) = \frac{\partial}{\partial r_{ij}}\left(\frac{\partial F_H}{\partial \rho_{H,i}}\right)\frac{\partial \rho_{H,i}}{\partial r_{ij}} + \frac{\partial F_H}{\partial \rho_{H,i}} \frac{\partial^2 \rho_{H,i}}{\partial r_{ij}^{\ 2}} = \frac{\partial^2 F_H}{\partial \rho_{H,i}^2}\left(\frac{\partial \rho_{H,i}}{\partial r_{ij}}\right)^2 + \frac{\partial F_H}{\partial \rho_{H,i}} \frac{\partial^2 \rho_{H,i}}{\partial r_{ij}^{\ 2}} \tag{17a}$$

$$\frac{\partial^2 F_I}{\partial r_{ij}^{\ 2}} = \frac{\partial^2 F_I}{\partial \rho_{I,i}^2}\left(\frac{\partial \rho_{I,i}}{\partial r_{ij}}\right)^2 + \frac{\partial F_I}{\partial \rho_{I,i}} \frac{\partial^2 \rho_{I,i}}{\partial r_{ij}^{\ 2}} \tag{17b}$$

$$\frac{\partial E_H}{\partial r_{ij}} = \frac{\partial F_H}{\partial r_{ij}} + \frac{1}{2}\sum_{\substack{j=1 \\ j\neq i}}^{N_H} \frac{\partial \phi_{H-H,ij}}{\partial r_{ij}} + \frac{1}{2}\sum_{\substack{j=1 \\ j\neq i}}^{N_I} \frac{\partial \phi_{H-I,ij}}{\partial r_{ij}} \tag{18a}$$

$$\frac{\partial E_I}{\partial r_{ij}} = \frac{\partial F_I}{\partial r_{ij}} + \frac{1}{2}\sum_{\substack{j=1 \\ j\neq i}}^{N_I} \frac{\partial \phi_{I-I,ij}}{\partial r_{ij}} + \frac{1}{2}\sum_{\substack{j=1 \\ j\neq i}}^{N_H} \frac{\partial \phi_{I-H,ij}}{\partial r_{ij}} \tag{18b}$$

$$\frac{\partial^2 E_H}{\partial r_{ij}^2} = \frac{\partial^2 F_H}{\partial r_{ij}^2} + \frac{1}{2}\sum_{\substack{j=1 \\ j\neq i}}^{N_H} \frac{\partial^2 \phi_{H-H,ij}}{\partial r_{ij}^2} + \frac{1}{2}\sum_{\substack{j=1 \\ j\neq i}}^{N_I} \frac{\partial^2 \phi_{H-I,ij}}{\partial r_{ij}^2} \tag{19a}$$

$$\frac{\partial^2 E_I}{\partial r_{ij}^2} = \frac{\partial^2 F_I}{\partial r_{ij}^2} + \frac{1}{2}\sum_{\substack{j=1 \\ j\neq i}}^{N_I} \frac{\partial^2 \phi_{I-I,ij}}{\partial r_{ij}^2} + \frac{1}{2}\sum_{\substack{j=1 \\ j\neq i}}^{N_H} \frac{\partial^2 \phi_{I-H,ij}}{\partial r_{ij}^2} \tag{19b}$$

$$\frac{\partial E_H}{\partial r_i} = \frac{\partial E_H}{\partial r_{ij}} \frac{\partial r_{ij}}{\partial r_i} = \frac{\partial E_H}{\partial r_{ij}} \frac{r_i}{r_{ij}} \tag{20a}$$

$$\frac{\partial E_I}{\partial r_i} = \frac{\partial E_I}{\partial r_{ij}} \frac{\partial r_{ij}}{\partial r_i} = \frac{\partial E_I}{\partial r_{ij}} \frac{r_i}{r_{ij}} \tag{20b}$$

$$\frac{\partial E_H}{\partial r_{\mathrm{j}}} = \frac{\partial E_H}{\partial r_{ij}} \frac{\partial r_{ij}}{\partial r_j} = \frac{\partial E_H}{\partial r_{ij}} \frac{r_j}{r_{ij}} \tag{21a}$$

$$\frac{\partial E_I}{\partial r_{\mathrm{j}}} = \frac{\partial E_I}{\partial r_{ij}} \frac{\partial r_{ij}}{\partial r_j} = \frac{\partial E_I}{\partial r_{ij}} \frac{r_j}{r_{ij}} \tag{21b}$$

$$\frac{\partial^2 E_H}{\partial r_i \partial r_j} = \frac{\partial}{\partial r_j}\left(\frac{\partial E_H}{\partial r_{ij}} \frac{\partial r_{ij}}{\partial r_i}\right) = \frac{\partial}{\partial r_j}\left(\frac{\partial E_H}{\partial r_{ij}}\right)\frac{\partial r_{ij}}{\partial r_i} + \frac{\partial E_H}{\partial r_{ij}} \frac{\partial}{\partial r_j}\left(\frac{\partial r_{ij}}{\partial r_i}\right)$$

$$= \frac{\partial}{\partial r_{ij}} \frac{\partial r_{ij}}{\partial r_j}\left(\frac{\partial E_H}{\partial r_{ij}}\right)\frac{\partial r_{ij}}{\partial r_i} + \frac{\partial E_H}{\partial r_{ij}} \frac{\partial}{\partial r_j}\left(\frac{\partial r_{ij}}{\partial r_i}\right)$$

$$= \left(\frac{\partial^2 E_H}{\partial r_{ij}^2}\right)\frac{\partial r_{ij}}{\partial r_i} \frac{\partial r_{ij}}{\partial r_j} + \frac{\partial E_H}{\partial r_{ij}}\left(\frac{\partial^2 r_{ij}}{\partial r_i \partial r_j}\right)$$

$$= \left(\frac{\partial^2 E_H}{\partial r_{ij}^2}\right)\frac{r_i r_j}{r_{ij}^2} + \frac{\partial E_H}{\partial r_{ij}}\left(\delta_{ij}\frac{1}{r_{ij}} - \frac{r_i r_j}{r_{ij}^3}\right) \tag{22a}$$

$$\frac{\partial^2 E_I}{\partial r_i \partial r_j} = \frac{\partial}{\partial r_j}\left(\frac{\partial E_I}{\partial r_{ij}} \frac{\partial r_{ij}}{\partial r_i}\right) = \frac{\partial}{\partial r_j}\left(\frac{\partial E_I}{\partial r_{ij}}\right)\frac{\partial r_{ij}}{\partial r_i} + \frac{\partial E_I}{\partial r_{ij}} \frac{\partial}{\partial r_j}\left(\frac{\partial r_{ij}}{\partial r_i}\right)$$

$$= \frac{\partial}{\partial r_{ij}} \frac{\partial r_{ij}}{\partial r_j}\left(\frac{\partial E_I}{\partial r_{ij}}\right)\frac{\partial r_{ij}}{\partial r_i} + \frac{\partial E_I}{\partial r_{ij}} \frac{\partial}{\partial r_j}\left(\frac{\partial r_{ij}}{\partial r_i}\right)$$

$$= \left(\frac{\partial^2 E_I}{\partial r_{ij}^2}\right)\frac{\partial r_{ij}}{\partial r_i} \frac{\partial r_{ij}}{\partial r_j} + \frac{\partial E_I}{\partial r_{ij}}\left(\frac{\partial^2 r_{ij}}{\partial r_i \partial r_j}\right)$$

$$= \left(\frac{\partial^2 E_I}{\partial r_{ij}^2}\right)\frac{r_i r_j}{r_{ij}^2} + \frac{\partial E_I}{\partial r_{ij}}\left(\delta_{ij}\frac{1}{r_{ij}} - \frac{r_i r_j}{r_{ij}^3}\right) \tag{22b}$$

MIXING RULE PAIR POTENTIAL DERIVATES

$$\phi_{ab}(r) = \frac{1}{2}\left[\frac{f_b(r)}{f_a(r)}\phi_{aa}(r) + \frac{f_a(r)}{f_b(r)}\phi_{bb}(r)\right] = \frac{1}{2}\left[f_b f_a{}^{-1}\phi_{aa} + f_a f_b{}^{-1}\phi_{bb}\right] \tag{23}$$

$$\frac{\partial^2\phi_{ab}}{\partial r^2} = \frac{1}{2}\frac{\partial}{\partial r}\left\{\begin{array}{l}\left[\left(-f_b\dot{f}_a f_a^{-2} + \dot{f}_b f_a^{-1}\right)\phi_{aa} + \left(f_b f_a^{-1}\right)\dot{\phi}_{aa}\right] + \\ \left[\left(-f_a\dot{f}_b f_b^{-2} + \dot{f}_a f_b^{-1}\right)\phi_{bb} + \left(f_a f_b^{-1}\right)\dot{\phi}_{bb}\right]\end{array}\right\} \tag{24}$$

$$\frac{\partial^2\phi_{ab}}{\partial r^2} = \frac{1}{2}\left\{\begin{array}{l}\left\{-1\left[\dot{f}_b\dot{f}_a f_a^{-2} + f_b\ddot{f}_a f_a^{-2} - 2f_b\dot{f}_a^2 f_a^{-3}\right] + \left[\ddot{f}_b f_a^{-1} - 1\dot{f}_b f_a^{-2}\dot{f}_a\right]\right\}\phi_{aa} \\ +\left(-f_b\dot{f}_a f_a^{-2} + \dot{f}_b f_a^{-1}\right)\dot{\phi}_{aa} + \dot{f}_b f_a^{-1}\dot{\phi}_{aa} - 1f_b f_a^{-2}\dot{f}_a\dot{\phi}_{aa} + f_b f_a^{-1}\ddot{\phi}_{aa} \\ + \left\{-1\left[\dot{f}_a\dot{f}_b f_b^{-2} + f_a\ddot{f}_b f_b^{-2} - 2f_a\dot{f}_b^2 f_b^{-3}\right] + \left[\ddot{f}_a f_b^{-1} - 1\dot{f}_a f_b^{-2}\dot{f}_b\right]\right\}\phi_{bb} \\ +\left(-f_a\dot{f}_b f_b^{-2} + \dot{f}_a f_b^{-1}\right)\dot{\phi}_{bb} + \dot{f}_a f_b^{-1}\dot{\phi}_{bb} - 1f_a f_b^{-2}\dot{f}_b\dot{\phi}_{bb} + f_a f_b^{-1}\ddot{\phi}_{bb}\end{array}\right\} \tag{25}$$

TOTAL ENERGY

$$\begin{aligned}E_{total} = &\sum_{i=1}^{N_H} F_{H,i}(\rho_{H,i}) + \frac{1}{2}\sum_{i=1}^{N_H}\sum_{\substack{j=1\\ j\neq i}}^{N_H}\phi_{H-H,ij}(r_{ij}) + \frac{1}{2}\sum_{i=1}^{N_H}\sum_{\substack{j=1\\ j\neq i}}^{N_I}\phi_{H-I,ij}(r_{ij}) + \\ &\sum_{i=1}^{N_I} F_{I,i}(\rho_{H,i}) + \frac{1}{2}\sum_{i=1}^{N_I}\sum_{\substack{j=1\\ j\neq i}}^{N_H}\phi_{I-H,ij}(r_{ij}) + \frac{1}{2}\sum_{i=1}^{N_I}\sum_{\substack{j=1\\ j\neq i}}^{N_H}\phi_{I-H,ij}(r_{ij})\end{aligned} \tag{26}$$

COHESIVE ENERGY

$$E_c = \frac{1}{N_H + N_I}\left[N_H E_H + N_I E_I\right] \tag{27}$$

$$E_c = \frac{1}{N_H + N_I}\left[\begin{array}{l}N_H\left(F_{H,i}(\rho_{H,i}) + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_H}\phi_{H-H,ij}(r_{ij}) + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_I}\phi_{H-I,ij}(r_{ij})\right) + \\ N_I\left(F_{I,i}(\rho_{H,i}) + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_H}\phi_{I-H,ij}(r_{ij}) + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_H}\phi_{I-H,ij}(r_{ij})\right)\end{array}\right] \tag{28}$$

$$= \frac{1}{\mathrm{x} + y}\left[\begin{array}{l} \mathrm{x}\left(F_{H,i}(\rho_{H,i}) + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_H}\phi_{H-H,ij}(r_{ij}) + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_I}\phi_{H-I,ij}(r_{ij})\right) + \\ y\left(F_{I,i}(\rho_{I,i}) + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_H}\phi_{I-H,ij}(r_{ij}) + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_I}\phi_{I-I,ij}(r_{ij})\right) \end{array}\right] \tag{29}$$

or with three types: A, B, and C

$$E_c = \frac{1}{\mathrm{x}+y+z}\left[\begin{array}{l} \mathrm{x}\left(F_{A,i}(\rho_{A,i}) + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_A}\varphi_{A-A,ij}(r_{ij}) + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_B}\varphi_{A-B,ij}(r_{ij}) + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_C}\varphi_{A-C,ij}(r_{ij})\right) + \\ y\left(F_{B,i}(\rho_{B,i}) + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_B}\varphi_{B-B,ij}(r_{ij}) + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_A}\varphi_{B-A,ij}(r_{ij}) + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_C}\varphi_{B-C,ij}(r_{ij})\right) + \\ z\left(F_{C,i}(\rho_{C,i}) + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_C}\varphi_{C-C,ij}(r_{ij}) + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_A}\varphi_{C-A,ij}(r_{ij}) + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_B}\varphi_{C-B,ij}(r_{ij})\right) \end{array}\right] \tag{30}$$

$$\rho_{\mathrm{A},i} = \rho_{A-A,i} + \rho_{A-B,i} + \rho_{A-C,i} = \sum_{\substack{j=1\\ j\neq i}}^{N_A} f_A(r_{ij}) + \sum_{\substack{j=1\\ j\neq i}}^{N_B} f_B(r_{ij}) + \sum_{\substack{j=1\\ j\neq i}}^{N_C} f_C(r_{ij}) \tag{31}$$

$$\rho_{\mathrm{B},i} = \rho_{B-A,i} + \rho_{B-B,i} + \rho_{B-C,i} = \sum_{\substack{j=1\\ j\neq i}}^{N_A} f_A(r_{ij}) + \sum_{\substack{j=1\\ j\neq i}}^{N_B} f_B(r_{ij}) + \sum_{\substack{j=1\\ j\neq i}}^{N_C} f_C(r_{ij}) \tag{32}$$

$$\rho_{\mathrm{C},i} = \rho_{C-A,i} + \rho_{C-B,i} + \rho_{C-C,i} = \sum_{\substack{j=1\\ j\neq i}}^{N_A} f_A(r_{ij}) + \sum_{\substack{j=1\\ j\neq i}}^{N_B} f_B(r_{ij}) + \sum_{\substack{j=1\\ j\neq i}}^{N_C} f_C(r_{ij}) \tag{33}$$

BULK MODULUS

$$B = \frac{\frac{\partial P}{\partial V}}{V_o} = V_o \frac{\partial P}{\partial V} = V_o \frac{\partial^2 E}{\partial V^2} \tag{34}$$

$$\frac{\partial E}{\partial V} = \frac{\partial}{\partial a}\frac{\partial a}{\partial V} E = \frac{\partial E}{\partial a}\frac{\partial a}{\partial V} \tag{34}$$

$$\frac{\partial^2 E}{\partial V^2} = \frac{\partial}{\partial V}\left(\frac{\partial E}{\partial a}\frac{\partial a}{\partial V}\right) = \frac{\partial}{\partial V}\left(\frac{\partial E}{\partial a}\right)\frac{\partial a}{\partial V} + \frac{\partial}{\partial V}\left(\frac{\partial a}{\partial V}\right)\frac{\partial E}{\partial a}$$

$$= \frac{\partial}{\partial V}\left(\frac{\partial E}{\partial a}\right)\frac{\partial a}{\partial V} + \frac{\partial}{\partial V}\left(\frac{\partial a}{\partial V}\right)\frac{\partial E}{\partial a}$$

$$= \frac{\partial}{\partial a}\frac{\partial a}{\partial V}\left(\frac{\partial E}{\partial a}\right)\frac{\partial a}{\partial V} + \frac{\partial}{\partial V}\left(\frac{\partial a}{\partial V}\right) * 0$$

$$= \frac{\partial^2 E}{\partial a^2}\left(\frac{\partial a}{\partial V}\right)^2 \tag{35}$$

$$B = V_o \frac{\partial^2 E}{\partial V^2} = a^3 \frac{\partial^2 E}{\partial a^2}\left(\frac{\partial a}{\partial V}\right)^2 = a^3 \frac{\partial^2 E}{\partial a^2}\left(\frac{1}{3a^2}\right)^2 = \frac{a^2}{9V_o}\frac{\partial^2 E}{\partial a^2} \tag{36}$$

$$= a^3 \frac{\partial^2 E}{\partial a^2}\left(\frac{1}{3a^2}\right)^2 = \frac{a^3}{9a^4}\frac{\partial^2 E}{\partial a^2} = \frac{a^3}{9Va}\frac{\partial^2 E}{\partial a^2} = \frac{a^2}{9V}\frac{\partial^2 E}{\partial a^2}$$

$$= \frac{a^2}{9V}\frac{\partial^2 E}{\partial a^2} = \frac{a^2}{9V}\frac{\partial^2 E_c}{\partial r_{ij}^{\,2}}\left(\frac{r_{ij}}{a}\right)^2$$

$$= \frac{a^2}{9V}\left(\frac{\partial^2 F}{\partial \mathrm{r}_{ij}^{\,2}} + \frac{1}{2}\sum_{j\neq i}\frac{\partial^2 \phi}{\partial r_{ij}^{\,2}}\right)\left(\frac{\mathrm{r}_{ij}}{a}\right)^2 \tag{37}$$

$$= \frac{a^2}{9V}\left(\frac{\partial^2 F}{\partial \rho^2}\left(\frac{\partial \rho}{\partial \mathrm{r}_{ij}}\right)^2 + \frac{\partial F}{\partial \rho}\frac{\partial^2 \rho}{\partial \mathrm{r}_{ij}^{\,2}} + \frac{1}{2}\sum_{j\neq i}\frac{\partial^2 \phi}{\partial r_{ij}^{\,2}}\right)\left(\frac{\mathrm{r}_{ij}}{a}\right)^2$$

$$= \frac{a^2}{9\Omega}\left[F''(\rho)\left[\frac{\partial\rho}{\partial r_{ij}}\frac{r_{ij}}{a}\right]^2 + F'(\rho)\left[\frac{\partial^2\rho}{\partial r_{ij}^2}\frac{r_{ij}^2}{a^2}\right] + \frac{1}{2}\sum_{i\neq j,j}\phi''_{ij}(r_{ij})\frac{r_{ij}^2}{a^2}\right] \quad (38)$$

SOLID SOLUTION BULK MODULUS

Substituting Equations (15a) , (15b), (17a), (17b) into (31) we get

$$B_I = \frac{a^2}{9V_o}\left[\frac{\partial^2 F_I}{\partial r_{ij}^2} + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_H}\frac{\partial^2\phi_{I-H,ij}}{\partial r_{ij}^2} + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_I}\frac{\partial^2\phi_{I-I,ij}}{\partial r_{ij}^2}\right]\left(\frac{r_{ij}}{a}\right)^2 \quad (39a)$$

$$B_H = \frac{a^2}{9V_o}\left[\frac{\partial^2 F_H}{\partial r_{ij}^2} + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_H}\frac{\partial^2\phi_{H-H,ij}}{\partial r_{ij}^2} + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_I}\frac{\partial^2\phi_{H-I,ij}}{\partial r_{ij}^2}\right]\left(\frac{r_{ij}}{a}\right)^2 \quad (39b)$$

$$\begin{aligned} B_T &= B_H + B_I \\ &= \frac{a^2}{9V_o}\left[\frac{\partial^2 F_H}{\partial r_{ij}^2} + \frac{\partial^2 F_I}{\partial r_{ij}^2} + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_H}\frac{\partial^2\phi_{H-H,ij}}{\partial r_{ij}^2} + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_I}\frac{\partial^2\phi_{H-I,ij}}{\partial r_{ij}^2} + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_H}\frac{\partial^2\phi_{I-H,ij}}{\partial r_{ij}^2} + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_I}\frac{\partial^2\phi_{I-I,ij}}{\partial r_{ij}^2}\right]\left(\frac{r_{ij}}{a}\right)^2 \\ &= \frac{a^2}{9V_o}\left[\frac{\partial^2 F_H}{\partial r_{ij}^2} + \frac{\partial^2 F_I}{\partial r_{ij}^2} + \frac{1}{2}\left(\sum_{\substack{j=1\\ j\neq i}}^{N_H}\frac{\partial^2\phi_{H-H,ij}}{\partial r_{ij}^2} + \sum_{\substack{j=1\\ j\neq i}}^{N_I}\frac{\partial^2\phi_{H-I,ij}}{\partial r_{ij}^2} + \sum_{\substack{j=1\\ j\neq i}}^{N_H}\frac{\partial^2\phi_{I-H,ij}}{\partial r_{ij}^2} + \sum_{\substack{j=1\\ j\neq i}}^{N_I}\frac{\partial^2\phi_{I-I,ij}}{\partial r_{ij}^2}\right)\right]\left(\frac{r_{ij}}{a}\right)^2 \end{aligned} \quad (40)$$

STRESS

$$\sigma_{H,ij} = \frac{1}{\Omega_o}\left[\frac{\partial E}{\partial r_i}r_j\right] \quad (41)$$

SOLID SOLUTION STRESS

Substituting Equation (20a) into (35), we get

$$\sigma_{H,ij} = \frac{1}{\Omega_o}\left[\frac{\partial E_H}{\partial r_{ij}}\frac{r_i r_j}{r_{ij}}\right] \tag{42}$$

$$\sigma_{H,ij} = \frac{1}{\Omega_o}\left[\frac{\partial F_H}{\partial \rho_{H,i}}\frac{\partial \rho_{H,i}}{\partial r_{ij}} + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_H}\frac{\partial \phi_{H-H,ij}}{\partial r_{ij}} + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_I}\frac{\partial \phi_{H-I,ij}}{\partial r_{ij}}\right]\frac{r_i r_j}{r_{ij}} \tag{43}$$

$$\sigma_{H,ij} = \frac{1}{\Omega_o}\left[\frac{\partial F_H}{\partial \rho_{H,i}}\left(\frac{\partial \rho_{H\text{-}H,i}}{\partial r_{ij}} + \frac{\partial \rho_{H-I,i}}{\partial r_{ij}}\right) + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_H}\frac{\partial \phi_{H-H,ij}}{\partial r_{ij}} + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_I}\frac{\partial \phi_{H-I,ij}}{\partial r_{ij}}\right]\frac{r_i r_j}{r_{ij}} \tag{44}$$

$$\sigma_{I,ij} = \frac{1}{\Omega_o}\left[\frac{\partial F_I}{\partial \rho_{I,i}}\left(\frac{\partial \rho_{I-I,i}}{\partial r_{ij}} + \frac{\partial \rho_{I-H,i}}{\partial r_{ij}}\right) + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_I}\frac{\partial \phi_{I-I,ij}}{\partial r_{ij}} + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_H}\frac{\partial \phi_{I-H,ij}}{\partial r_{ij}}\right]\frac{r_i r_j}{r_{ij}} \tag{45}$$

$$\begin{aligned}\sigma_{T,ij} &= \frac{1}{\Omega_o}\left[\frac{\partial F_H}{\partial \rho_{H,i}}\left(\frac{\partial \rho_{H\text{-}H,i}}{\partial r_{ij}} + \frac{\partial \rho_{H-I,i}}{\partial r_{ij}}\right) + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_H}\frac{\partial \phi_{H-H,ij}}{\partial r_{ij}} + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_I}\frac{\partial \phi_{H-I,ij}}{\partial r_{ij}}\right]\frac{r_i r_j}{r_{ij}} + \\ &= \frac{1}{\Omega_o}\left[\frac{\partial F_I}{\partial \rho_{I,i}}\left(\frac{\partial \rho_{I-I,i}}{\partial r_{ij}} + \frac{\partial \rho_{I-H,i}}{\partial r_{ij}}\right) + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_I}\frac{\partial \phi_{I-I,ij}}{\partial r_{ij}} + \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{N_H}\frac{\partial \phi_{I-H,ij}}{\partial r_{ij}}\right]\frac{r_i r_j}{r_{ij}}\end{aligned} \tag{46}$$

ELASTIC CONSTANTS

$$C_{\mathrm{ijkl}} = \frac{1}{\Omega_e}\frac{\partial^2 E}{\partial r_i \partial r_j} r_k r_l \tag{47}$$

Substituting Equations (18a) and (19a) into (41) and Equations (18b) and (19b) into (41), we get

$$C_{\mathrm{ijkl,H}} = \frac{1}{\Omega_e}\left[\left(\frac{r_i r_j}{r_{ij}^2}\right)\frac{\partial^2 E_H}{\partial r_{ij}^2} + \left(\delta_{ik}\frac{1}{r_{ij}} - \frac{r_i r_j}{r_{ij}^3}\right)\frac{\partial E_H}{\partial r_{ij}}\right] r_k r_l \tag{48a}$$

$$C_{\mathrm{ijkl,I}} = \frac{1}{\Omega_e}\left[\left(\frac{r_i r_j}{r_{ij}^2}\right)\frac{\partial^2 E_I}{\partial r_{ij}^2} + \left(\delta_{ik}\frac{1}{r_{ij}} - \frac{r_i r_j}{r_{ij}^3}\right)\frac{\partial E_I}{\partial r_{ij}}\right] r_k r_l \tag{48b}$$

Substituting Equations (20) and (22) into (47) and Equations (21) and (23) into (48), we get

$$C_{\mathrm{ijkl,H}} = \frac{1}{\Omega_{\mathrm{e}}}\left[\frac{\partial^2 F_H}{\partial \rho_{u,i}^2}\left(V_{\mathrm{ij}}^{H-H} + V_{\mathrm{ij}}^{H-I}\right)\left(V_{\mathrm{kl}}^{H-H} + V_{\mathrm{kl}}^{H-I}\right) + \frac{\partial F_H}{\partial \rho_{H,i}}\left(W_{\mathrm{ijkl}}^{H-H} + W_{\mathrm{ijkl}}^{H-I}\right) + B_{\mathrm{ijkl}}^{H-H} + B_{\mathrm{ijkl}}^{H-I}\right] \tag{49a}$$

$$C_{\mathrm{ijkl,I}} = \frac{1}{\Omega_{\mathrm{e}}}\left[\frac{\partial^2 F_I}{\partial \rho_{u,i}^2}\left(V_{\mathrm{ij}}^{I-I} + V_{\mathrm{ij}}^{I-H}\right)\left(V_{\mathrm{kl}}^{I-I} + V_{\mathrm{kl}}^{I-H}\right) + \frac{\partial F_H}{\partial \rho_{H,i}}\left(W_{\mathrm{ijkl}}^{I-I} + W_{\mathrm{ijkl}}^{I-H}\right) + B_{\mathrm{ijkl}}^{I-I} + B_{\mathrm{ijkl}}^{I-H}\right] \tag{49b}$$

where

$$W_{\mathrm{ijkl}}^{H} = \left(\sum_{\substack{j=1 \\ j\neq i}}^{N_H} \frac{\partial^2 \rho_{H,i}}{\partial r_{ij}^{\,2}} - \sum_{\substack{j=1 \\ j\neq i}}^{N_{\mathrm{H}}} \frac{1}{\mathrm{r}_{ij}} \frac{\partial \rho_{H,i}}{\partial r_{ij}}\right) \frac{r_i r_j r_k r_l}{r_{ij}^2} + \delta_{ik} \frac{r_k r_l}{r_{ij}} \left(\frac{\partial \rho_{H,i}}{\partial r_{ij}}\right) \tag{50a}$$

$$W_{\mathrm{ijkl}}^{I} = \left(\sum_{\substack{j=1 \\ j\neq i}}^{N_{\mathrm{I}}} \frac{\partial^2 \rho_{I,i}}{\partial r_{ij}^{\,2}} - \sum_{\substack{j=1 \\ j\neq i}}^{N_I} \frac{1}{\mathrm{r}_{ij}} \frac{\partial \rho_{I,i}}{\partial r_{ij}}\right) \frac{r_i r_j r_k r_l}{r_{ij}^2} + \delta_{ik} \frac{r_k r_l}{r_{ij}} \left(\frac{\partial \rho_{I,i}}{\partial r_{ij}}\right) \tag{50b}$$

$$B_{\mathrm{ijkl}}^{H} = \frac{1}{2}\left[\sum_{\substack{j=1 \\ j\neq i}}^{N_H} \frac{\partial^2 \phi_{H,ij}}{\partial r_{ij}^2} - \frac{1}{r_{ij}} \sum_{\substack{j=1 \\ j\neq i}}^{N_H} \frac{\partial \phi_{H,ij}}{\partial r_{ij}}\right] \frac{r_i r_j r_k r_l}{r_{ij}^2} + \frac{1}{2}\delta_{ik}\left(\sum_{\substack{j=1 \\ j\neq i}}^{N_H} \frac{\partial \phi_{H,ij}}{\partial r_{ij}}\right) \frac{r_k r_l}{r_{ij}} \tag{50a}$$

$$B_{\mathrm{ijkl}}^{I} = \frac{1}{2}\left[\sum_{\substack{j=1 \\ j\neq i}}^{N_I} \frac{\partial^2 \phi_{I-I,ij}}{\partial r_{ij}^2} - \frac{1}{r_{ij}} \sum_{\substack{j=1 \\ j\neq i}}^{N_I} \frac{\partial \phi_{I-I,ij}}{\partial r_{ij}}\right] \frac{r_i r_j r_k r_l}{r_{ij}^2} + \frac{1}{2}\delta_{ik}\left(\sum_{\substack{j=1 \\ j\neq i}}^{N_I} \frac{\partial \phi_{I-I,ij}}{\partial r_{ij}}\right) \frac{r_k r_l}{r_{ij}} \tag{50b}$$

$$V^{H-H}{}_{ij} = \left(\frac{\partial \rho_{H\text{-}H,i}}{\partial r_{ij}}\right) \frac{r_i r_j}{r_{ij}}, \qquad V^{I-I}{}_{ij} = \left(\frac{\partial \rho_{I\text{-}I,i}}{\partial r_{ij}}\right) \frac{r_i r_j}{r_{ij}} \tag{51a}$$

$$V^{H-H}{}_{kl} = \left(\frac{\partial \rho_{H\text{-}H,i}}{\partial r_{ij}}\right) \frac{r_k r_l}{r_{ij}}, \qquad V^{I-I}{}_{kl} = \left(\frac{\partial \rho_{I\text{-}I,i}}{\partial r_{ij}}\right) \frac{r_k r_l}{r_{ij}} \tag{51b}$$

VACANCY FORMATION ENERGY

$$E_{UV}^{1} = -\frac{1}{2}\sum_{m}\phi(\mathrm{r}_{m}) - \sum_{m}F(\bar{\rho}) + \sum_{m}F(\bar{\rho} - f(r_{m})) \tag{52}$$

$$E_{V}^{1} = -\frac{1}{2}\sum_{m}\phi(\mathrm{r}_{m}) - \sum_{m}F(\bar{\rho}) + \sum_{m}F(\bar{\rho} - f(r_{m})) + E_{\mathrm{relax}} \tag{53}$$

$$E_{UV}^{2} = -2E_{V}^{1} + \phi(r_{e}) - \sum_{m}\phi(\mathrm{r}_{m}) - \sum_{m}F(\bar{\rho}) + \sum_{m>2}F(\bar{\rho} - 2f(r_{m})) \tag{54}$$

GIBBS FREE ENERGY MIXING

$$\Delta\mathrm{G}^{\mathrm{mix}} = \Delta\mathrm{H}^{\mathrm{mix}} - \Delta\mathrm{S}^{\mathrm{mix}} \cdot \mathrm{T} \tag{55}$$

$$\left[\mathrm{N}_{\mathrm{Pd}} \cdot x\right]PdH + \left[N_{Pd} \cdot \left(1 - x\right)\right]Pd \rightarrow \left[N_{Pd}\right]PdH_{x} \tag{56}$$

$$\Delta E = N_{Pd}\left(1 + x\right) \cdot E_{PdH_{x}} - 2 \cdot N_{Pd} \cdot x \cdot E_{PdH} - N_{Pd} \cdot (1 - x)E_{Pd} \tag{57}$$

$$\Delta E = \left[\left(1 + x\right) \cdot E_{PdH_{x}} - 2 \cdot \mathrm{x} \cdot E_{PdH} - (1 - x)E_{Pd}\right] / (1 - x) \tag{58}$$

Replacing x with mol fraction X=x/(1+x), obtain the heat of mixing

$$\Delta\mathrm{H}^{\mathrm{mix}} = E_{PdHx} - 2X \cdot E_{PdH} - \left(1 - 2X\right) \cdot E_{Pd} \tag{59}$$

$$\Delta S_{t} = -N_{Pd} \cdot k_{B} \cdot \left[x \cdot \ln\left(x\right) + \left(1 - x\right) \cdot \ln\left(x\right)\right] \tag{60}$$

$$\Delta S = -k_{B} \cdot \left[x \cdot \ln\left(x\right) + \left(1 - x\right) \cdot \ln\left(1 - x\right)\right] / \left(1 + x\right) \tag{61}$$

$$\Delta S_{t} = -k_{B} \cdot \left[X \cdot \ln\left[X / \left(1 - X\right)\right] + \left(1 - 2 \cdot X\right) \cdot \ln\left[(1 - 2X) / \left(1 - X\right)\right]\right] \tag{62}$$

APPENDIX C:

EAM CODE VALIDATION

Comparison with Zhou et al.

To validate the implementation of the property equations used in this research for the Pd-H system, we implemented the EAM model potentials and fitting data of Zhou et al. in our MATLAB EAM code [4]. In their more comprehensive report from Sandia National Laboratory, Zhou et al. give their Mathematic fitting code and the results obtained from fitting [49]. The reported results are in excellent agreement with the calculated properties obtained with the MATLAB EAM code used in this research. Figures 31 and 32 show that the cohesive energy calculated for both OC and TE structures match the reported results. The MATLAB code also matched the elastic properties reported. Figures 33 and 34 show that the bulk modulus and shear elastic constant C' are in exact agreement with the published results.

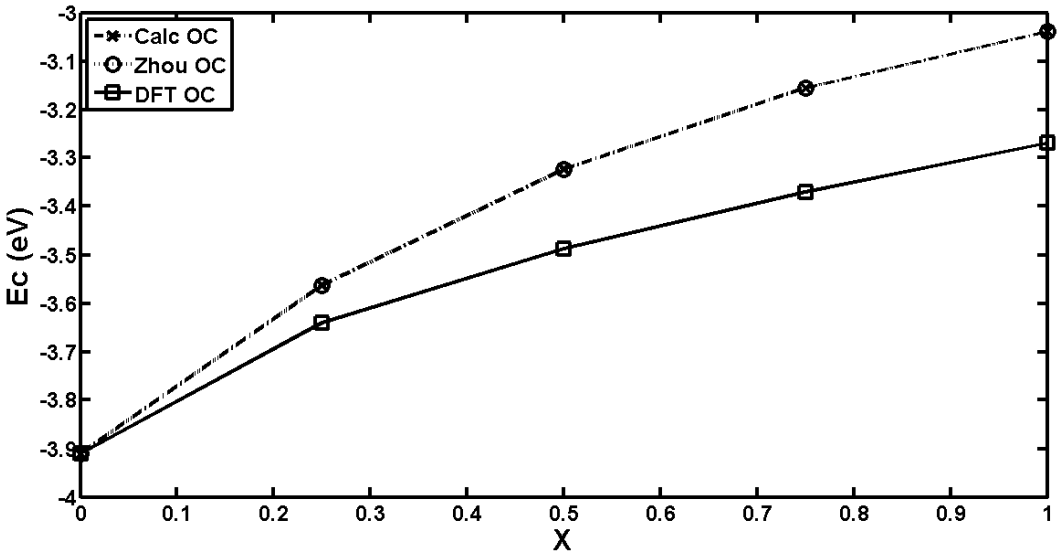

Figure 31. Cohesive energy for OC structure

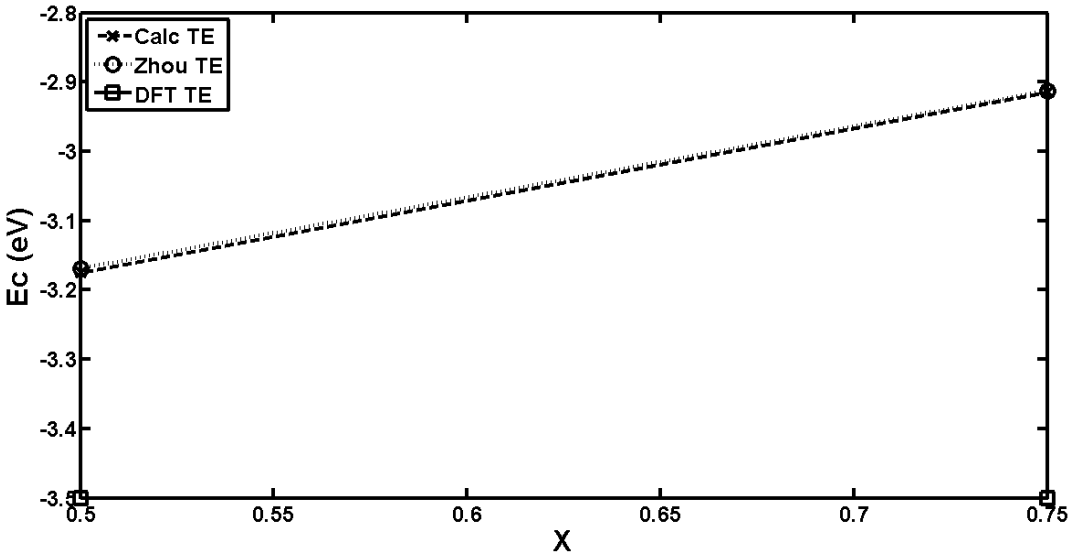

Figure 32. Cohesive energy for TE structure

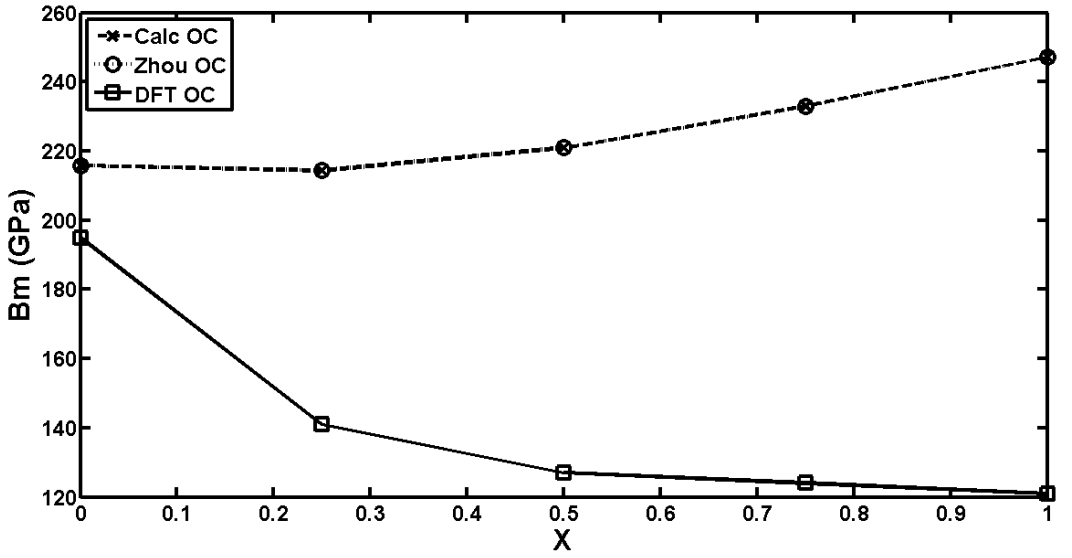

Figure 33. Bulk modulus for OC structure

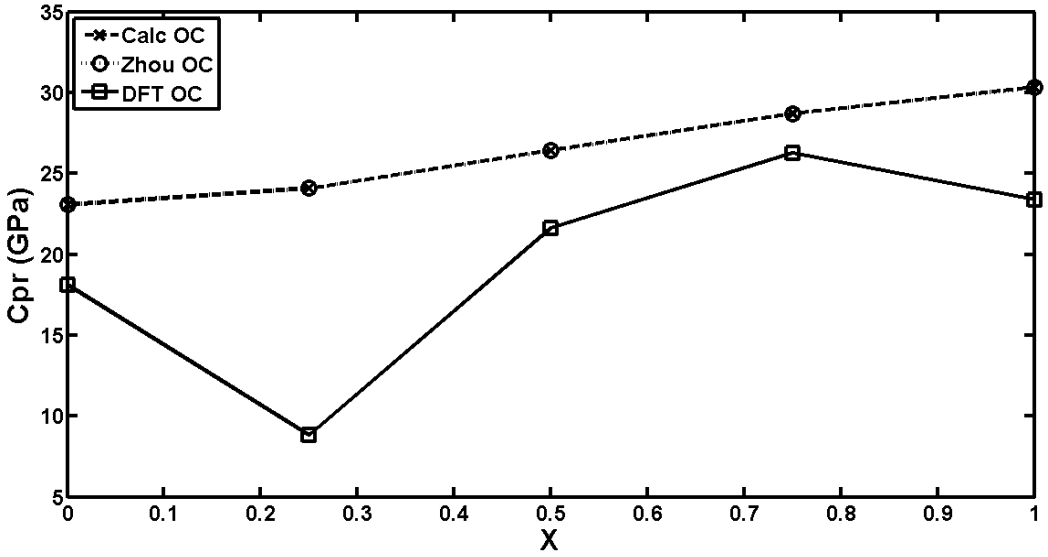

Figure 34. C' for OC structure

Central Atom Method comparison with MD

To validate the MATLAB code that generates the potential file in DYNAMO format, as well as to validate the central atom calculations with three types, the results from fitting were compared with MD runs done for zero steps at the desired lattice constant in LAMMPS. The results were compared for all of the structures used in fitting. The EAM calculations were in exact agreement with MD; showing the validity of the central atom calculations in MATLAB for representing the bulk system.

Table 12. Central Atom Method comparison with MD

Composition	Structure	a (A)	MATLAB E_c (eV)	MD E_c (eV)
$Pd_{0.75}Ag_{0.25}$	FCC	3.9400	-3.66598	-3.66598
$Pd_{0.75}Ag_{0.25}H_{0.25}$	OC1	4.0042	-3.44149	-3.44149
$Pd_{0.75}Ag_{0.25}H_{0.25}$	OC2	4.0056	-3.40279	-3.40279
$Pd_{0.75}Ag_{0.25}H_{0.25}$	TE1	4.0016	-3.39306	-3.39306
$Pd_{0.75}Ag_{0.25}H_{0.50}$	OC1 OC2	4.0663	-3.29534	-3.29534
$Pd_{0.75}Ag_{0.25}H_{0.50}$	OC2 OC2	4.0800	-3.26530	-3.26530
$Pd_{0.75}Ag_{0.25}H_{0.50}$	TE1 TE2	4.1239	-3.15743	-3.15743
$Pd_{0.75}Ag_{0.25}H_{0.75}$	OC1 OC2 OC2	4.1277	-3.23853	-3.23853
$Pd_{0.75}Ag_{0.25}H_{0.75}$	OC2 OC2 OC2	4.1315	-3.21596	-3.21596
$Pd_{0.75}Ag_{0.25}H_{0.75}$	TE1 TE2 TE3	4.2027	-3.04931	-3.04931
$Pd_{0.75}Ag_{0.25}H_{1.00}$	OC1 OC2 OC2 OC2	4.1848	-3.24563	-3.24563
$Pd_{0.75}Ag_{0.25}H_{1.00}$	TE1 TE2 TE3 TE4	4.2874	-3.02112	-3.02112

APPENDIX D:

CODE USED TO CALCULATE DATA IN THE THESIS

LAMMPS CODE

ELASTIC CONSTANTS CALCULATION [43]

The calculation of lattice constants, cohesive energies, and elastic constants can all be done with one test in LAMMPS. The in.elastic file uses additional files in the same directory for certain functions like designating the structure, EAM potential, etc.

in.elastic_PdAgH0886

```
# Compute elastic constant tensor for a crystal
#
# Written by Aidan Thompson (Sandia, athomps@sandia.gov)
#
#  This script uses the following three include files.
#
#   init.mod      (must be modified for different crystal structures)
#                 Define units, deformation parameters and initial
#            configuration of the atoms and simulation cell.
#
#
#   potential.mod    (must be modified for different pair styles)
#                Define pair style and other attributes
#                not stored in restart file
#
#
#   displace.mod    (displace.mod should not need to be modified)
#               Perform positive and negative box displacements
#               in direction ${dir} and size ${up}.
#               It uses the resultant changes
#               in stress to compute one
#               row of the elastic stiffness tensor
#
#               Inputs variables:
#                    dir = the Voigt deformation component
#                          (1,2,3,4,5,6)
#               Global constants:
#                    up = the deformation magnitude (strain units)
#                    cfac = conversion from LAMMPS pressure units to
#                    output units for elastic constants
#
#
#  To run this on a different system, it should only be necessary to
#  modify the files init.mod and potential.mod. In order to calculate
```

```
#  the elastic constants correctly, care must be taken to specify
#  the correct units in init.mod (units, cfac and cunits). It is also
#  important to verify that the minimization of energy w.r.t atom
#  positions in the deformed cell is fully converged.
#  One indication of this is that the elastic constants are
insensitive
#  to the choice of the variable ${up} in init.mod. Another is to
check
#  the final max and two-norm forces reported in the log file. If you
know
#  that minimization is not required, you can set maxiter = 0.0 in
#  init.mod.
#
#  There are two alternate versions of displace.mod provided.
#  They are displace_restart.mod and displace_reverse.mod.
#  The former resets the box using a restart file while
#  the latter reverses the deformation. Copy whichever
#  one you like best to displace.mod.
#

include init.mod-PdAgH-0.886
include potential.mod
include NPT.mod
# Compute initial state
fix 3 all box/relax  iso 0.0
minimize ${etol} ${ftol} ${maxiter} ${maxeval}

variable tmp equal pxx
variable pxx0 equal ${tmp}
variable tmp equal pyy
variable pyy0 equal ${tmp}
variable tmp equal pzz
variable pzz0 equal ${tmp}
variable tmp equal pyz
variable pyz0 equal ${tmp}
variable tmp equal pxz
variable pxz0 equal ${tmp}
variable tmp equal pxy
variable pxy0 equal ${tmp}

variable tmp equal lx
variable lx0 equal ${tmp}
variable tmp equal ly
variable ly0 equal ${tmp}
variable tmp equal lz
variable lz0 equal ${tmp}

# These formulas define the derivatives w.r.t. strain components
# Constants uses $, variables use v_
variable d1 equal -(v_pxx1-${pxx0})/(v_delta/v_len0)*${cfac}
variable d2 equal -(v_pyy1-${pyy0})/(v_delta/v_len0)*${cfac}
variable d3 equal -(v_pzz1-${pzz0})/(v_delta/v_len0)*${cfac}
```

```
variable d4 equal -(v_pyz1-${pyz0})/(v_delta/v_len0)*${cfac}
variable d5 equal -(v_pxz1-${pxz0})/(v_delta/v_len0)*${cfac}
variable d6 equal -(v_pxy1-${pxy0})/(v_delta/v_len0)*${cfac}

# Write restart
unfix 3
write_restart restart.equil

# uxx Perturbation

variable dir equal 1
include displace.mod

# uyy Perturbation

variable dir equal 2
include displace.mod

# uzz Perturbation

variable dir equal 3
include displace.mod

# uyz Perturbation

variable dir equal 4
include displace.mod

# uxz Perturbation

variable dir equal 5
include displace.mod

# uxy Perturbation

variable dir equal 6
include displace.mod

# Output final values

variable C11all equal ${C11}
variable C22all equal ${C22}
variable C33all equal ${C33}

variable C12all equal 0.5*(${C12}+${C21})
variable C13all equal 0.5*(${C13}+${C31})
variable C23all equal 0.5*(${C23}+${C32})

variable C44all equal ${C44}
variable C55all equal ${C55}
variable C66all equal ${C66}
```

```
variable C14all equal 0.5*(${C14}+${C41})
variable C15all equal 0.5*(${C15}+${C51})
variable C16all equal 0.5*(${C16}+${C61})

variable C24all equal 0.5*(${C24}+${C42})
variable C25all equal 0.5*(${C25}+${C52})
variable C26all equal 0.5*(${C26}+${C62})

variable C34all equal 0.5*(${C34}+${C43})
variable C35all equal 0.5*(${C35}+${C53})
variable C36all equal 0.5*(${C36}+${C63})

variable C45all equal 0.5*(${C45}+${C54})
variable C46all equal 0.5*(${C46}+${C64})
variable C56all equal 0.5*(${C56}+${C65})
variable Bm     equal (1/3)*(${C11all}+2*(${C12all}))
variable Cpr    equal 0.5*(${C11all}-(${C12all}))

# For Stillinger-Weber silicon, the analytical results
# are known to be (E. R. Cowley, 1988):
#               C11 = 151.4 GPa
#               C12 = 76.4 GPa
#               C44 = 56.4 GPa

print "Elastic Constant C11all = ${C11all} ${cunits}"
print "Elastic Constant C22all = ${C22all} ${cunits}"
print "Elastic Constant C33all = ${C33all} ${cunits}"

print "Elastic Constant C12all = ${C12all} ${cunits}"
print "Elastic Constant C13all = ${C13all} ${cunits}"
print "Elastic Constant C23all = ${C23all} ${cunits}"

print "Elastic Constant C44all = ${C44all} ${cunits}"
print "Elastic Constant C55all = ${C55all} ${cunits}"
print "Elastic Constant C66all = ${C66all} ${cunits}"

print "Elastic Constant C14all = ${C14all} ${cunits}"
print "Elastic Constant C15all = ${C15all} ${cunits}"
print "Elastic Constant C16all = ${C16all} ${cunits}"

print "Elastic Constant C24all = ${C24all} ${cunits}"
print "Elastic Constant C25all = ${C25all} ${cunits}"
print "Elastic Constant C26all = ${C26all} ${cunits}"

print "Elastic Constant C34all = ${C34all} ${cunits}"
print "Elastic Constant C35all = ${C35all} ${cunits}"
print "Elastic Constant C36all = ${C36all} ${cunits}"

print "Elastic Constant C45all = ${C45all} ${cunits}"
print "Elastic Constant C46all = ${C46all} ${cunits}"
print "Elastic Constant C56all = ${C56all} ${cunits}"
```

```
variable natoms equal "count(all)"
variable xlength equal "lx/4"
variable ecoh equal "pe/v_natoms"

print ""
print "Number of atoms = ${natoms};"
print "Lattice constant in x (Angstoms) = ${xlength};"
print "Cohesive energy (eV) = ${ecoh};"
print "Elastic Constant C11all = ${C11all} ${cunits}"
print "Elastic Constant C12all = ${C12all} ${cunits}"
print "Elastic Constant C44all = ${C44all} ${cunits}"
print "Elastic Constant Cprall = ${Cpr}    ${cunits}"
print "Bulk Modulus BMall      = ${Bm}     ${cunits}"

print "${xlength}     ${ecoh}    ${C11all}  ${C12all}  ${C44all}
     ${Cpr}     ${Bm}" file MD_results_PdAgH0886.dat
```

init.mod-PdAgH-0.886

```
# NOTE: This script can be modified for different atomic structures,
# units, etc. See in.elastic for more info.
#

# Define the finite deformation size. Try several values of this
# variable to verify that results do not depend on it.
variable up equal 1.0e-6

# Uncomment one of these blocks, depending on what units
# you are using in LAMMPS and for output

# metal units, elastic constants in eV/A^3
#units          metal
#variable cfac equal 6.2414e-7
#variable cunits string eV/A^3

# metal units, elastic constants in GPa
units      metal
variable cfac equal 1.0e-4
variable cunits string GPa

# real units, elastic constants in GPa
#units          real
#variable cfac equal 1.01325e-4
#variable cunits string GPa

# Define minimization parameters
variable etol equal 0.0
variable ftol equal 1.0e-10
variable maxiter equal 10000
variable maxeval equal 1000000
variable dmax equal 1.0e-6
```

```
# generate the box and atom positions using a diamond lattice
variable a equal 3.89
boundary   p p p
lattice         custom $a a1 1.0 0.0 0.0 a2 0.0 1.0 0.0 a3 0.0 0.0 1.0
basis 0.0 0.0 0.0 basis 0.5 0.5 0.0 basis 0.5 0.0 0.5 basis 0.0 0.5
0.5 &
basis 0.5 0.0 0.0 basis 0.0 0.5 0.0 basis 0.0 0.0 0.5 basis 0.5 0.5
0.5
#lattice        fcc $a
region          box prism 0 4.0 0 4.0 0 4.0 0 0 0
create_box 3 box
create_atoms    1 region box basis 1 2 basis 5 3 basis 6 3 basis 7 3
basis 8 3

group           vacancy id 5:229:8  # PdH0.885

delete_atoms    group vacancy


# Need to set mass to something, just to satisfy LAMMPS
mass 1 1.0e-20
```

potential.mod

```
# NOTE: This script can be modified for different pair styles
# See in.elastic for more info.

#include  init.mod

# Choose potential
pair_style     eam/alloy
pair_coeff     * * PdAgH.eam.alloy Pd Ag H

# Setup neighbor style
neighbor 1.0 nsq
neigh_modify once no every 1 delay 0 check yes

# Setup minimization style
min_style       cg
min_modify      dmax ${dmax} line quadratic

dump        1  all xyz 1000 PdAgH.xyz
dump        2  all custom 1000 PdAgH_Custom.xyz type id x y z
dump_modify 2  sort  1

# Setup output
thermo          1000

#  pxx,pyy,pzz,pxy,pxz,pyz = 6 components of pressure tensor
```

```
thermo_style custom step temp pe press pxx pyy pzz pxy pxz pyz lx ly
lz vol
thermo_modify norm no
```

NPT.mod

```
variable                t equal 500
variable                p equal 0
velocity                all create $t 4928459
#fix                    1 all npt temp $t $t 1 iso $p $p 100

fix                     1 all npt temp $t 1.0 10 iso 0.0 0.0 100
fix                     2 all temp/rescale 1 $t 1.0 0.01 1.0

run                     100000
unfix                   1
unfix                   2
```

FREE ENERGY AT 300 K

PdAgH-0292-NPT.in

```
units               metal
boundary            p p p
atom_style          atomic
variable            a equal 3.89

lattice             custom $a a1 1.0 0.0 0.0 a2 0.0 1.0 0.0 a3 0.0 0.0
1.0 basis 0.0 0.0 0.0 basis 0.5 0.5 0.0 basis 0.5 0.0 0.5 basis 0.0
0.5 0.5 basis 0.25 0.25 0.25 basis 0.75 0.75 0.75 basis 0.75 0.25 0.25
basis 0.25 0.75 0.25
#
region              box block 0 4 0 4 0 4
create_box          3 box
create_atoms        1 box basis 1 2 basis 5 3 basis 6 3 basis 7 3
basis 8 3
#
group           group1 id 5:509:8   # PdH0.784
group           group2 id 7:511:8
group           group3 id 6:422:8

delete_atoms    group  group1
delete_atoms    group  group2
delete_atoms    group  group3
#
pair_style          eam/alloy
#
pair_coeff          * * PdAgH.eam.alloy Pd Ag H
#
group               Pd type 1
group               Ag  type 2
```

```
group               H  type 3
#
neighbor            0.3 bin
neigh_modify        delay 5
variable            t index 300
variable            p index 0
#
velocity            all create $t 4928459
fix                 1 all npt temp $t 1 10 iso 0.0 0.0 100
fix                 2 all temp/rescale 1 $t $t 0.01 1.0
thermo_style        custom step temp press vol epair lx ly lz
thermo              100
dump                1 all xyz 100000 PdAgH-0292-NPT.xyz
dump                2 all xyz 100000 PdAgH-0292-SD.xyz
log                 PdAgH_0292.lammps
run                 100000

variable z loop 10

label loopz

run 100
variable natoms equal "count(all)"
variable xlength equal "lx/4"
variable ecoh equal "pe/v_natoms"
print "${xlength}     ${ecoh}" file MD_results_PdAgH0292_$z.dat

next       z
jump       SELF loopz
```

BULK MODULUS CALCULATIONS FOR $PdH_{0.185}$

PdAgH_0185_BM.in

```
units               metal
boundary            p p p
atom_style          atomic
variable            a equal 3.89

lattice                 custom $a a1 1.0 0.0 0.0 a2 0.0 1.0 0.0 a3 0.0
0.0 1.0 basis 0.0 0.0 0.0 basis 0.5 0.5 0.0 basis 0.5 0.0 0.5 basis
0.0 0.5 0.5 basis 0.5 0.5 0.5 basis 0.5 0.0 0.0 basis 0.0 0.5 0.0
basis 0.0 0.0 0.5
#
region              box block 0 4 0 4 0 4
create_box          3 box
create_atoms        1 box basis 1 2 basis 5 3 basis 6 3 basis 7 3
basis 8 3
#
group           group1 id 5:509:8   # PdH0.185
```

```
group           group2 id 7:511:8
group           group3 id 6:512:8
group           group4 id 8:136:8

delete_atoms    group  group1
delete_atoms    group  group2
delete_atoms    group  group3
delete_atoms    group  group4

#
pair_style              eam/alloy
pair_coeff              * * PdAgH.eam.alloy Pd Ag H
neighbor                0.3 bin
neigh_modify            delay 5
#
variable                n loop 21
variable                p index -10000 -9000 -8000 -7000 -6000 -5000 -
4000 -3000 -2000 -1000 0 1000 2000 3000 4000 5000 6000 7000 8000 9000
10000
#
# annealling step
velocity               all create 1000 4928459
fix                   1 all npt temp 1000 1.0 10 iso 0.0 0.0 100
fix                   2 all temp/rescale 1 1000 1.0 0.01 1.0
run                     10000
unfix                   1
unfix                   2
#

#
variable                t index 0.0001
velocity                all create $t 4928459

fix                     3 all npt temp $t $t 1 iso $p $p 100
fix                     4 all temp/rescale 1 $t $t 0.01 1.0



#
#thermo_style            custom press vol
#thermo                 1000
#dump                   1 all xyz 1000 $p_Pd.xyz
#
#log                     $p_Pd.lammps
#
run                     10000

variable pressure equal "press"
variable volume equal "vol"
print "${pressure}    ${volume}" file Press_Vol_$p_PdAgH0185.dat

clear
```

```
next                    p
next                    n
jump                    PdAgH_0185_BM.in
```

MATLAB PROGRAM FOR FITTING THE MODEL PARAMETERS

The fitting procedure was run from the MATLAB file Fit.m The objective function used in optimization was in a separate file objfunc.m Another file, Check_Fit.m, was used to generate plots and review the quality of the fit with the parameters obtained.

Fit.m

```
function EAM_Parameters_Fitting

    global a_PdAgH025_o1;
    global a_PdAgH025_o2;
    global a_PdAgH025_T;
    global a_PdAgH050_O1_O2;
    global a_PdAgH050_O2_O2;
    global a_PdAgH050_T;
    global a_PdAgH075_O1_O2_O2;
    global a_PdAgH075_O2_O2_O2;
    global a_PdAgH075_T;
    global a_PdAgH100_O1_O2_O2_O2;
    global a_PdAgH100_T;


    global Ec_exp_PdAgH025_O1;
    global Ec_exp_PdAgH025_O2;
    global Ec_exp_PdAgH025_T;
    global Ec_exp_PdAgH050_O1_O2;
    global Ec_exp_PdAgH050_O2_O2;
    global Ec_exp_PdAgH050_T;
    global Ec_exp_PdAgH075_O1_O2_O2;
    global Ec_exp_PdAgH075_O2_O2_O2;
    global Ec_exp_PdAgH075_T;
    global Ec_exp_PdAgH100_O1_O2_O2_O2;
    global Ec_exp_PdAgH100_T;


    global a_PdAgH025_TE1
    global a_PdAgH025_TE6
    global a_PdAgH025_TE8
    global a_PdAgH050_TE1_TE2
    global a_PdAgH050_TE3_TE4
    global a_PdAgH050_TE3_TE6
    global a_PdAgH050_TE1_TE8
    global a_PdAgH075_TE1_TE2_TE3
    global a_PdAgH075_TE1_TE2_TE8
    global a_PdAgH075_TE5_TE6_TE7
    global a_PdAgH100_TE1_TE2_TE3_TE4
    global a_PdAgH100_TE1_TE2_TE7_TE8
```

```
    global a_PdAgH100_TE1_TE5_TE6_TE7

    global Ec_exp_PdAgH025_TE1
    global Ec_exp_PdAgH050_TE3_TE4
    global Ec_exp_PdAgH050_TE1_TE8
    global Ec_exp_PdAgH075_TE1_TE2_TE3
    global Ec_exp_PdAgH075_TE1_TE2_TE8
    global Ec_exp_PdAgH075_TE5_TE6_TE7
    global Ec_exp_PdAgH100_TE1_TE2_TE3_TE4
    global Ec_exp_PdAgH100_TE1_TE2_TE7_TE8
    global Ec_exp_PdAgH100_TE1_TE5_TE6_TE7


% % %     spin polarized 18x18x18 kgrid  CG results factor shifted
    a_PdAgH025_o1 = 4.0042;
    a_PdAgH025_o2 = 4.0056;

    a_PdAgH050_O1_O2 = 4.0663;
    a_PdAgH050_O2_O2 = 4.0800;

    a_PdAgH075_O1_O2_O2 = 4.1277;
    a_PdAgH075_O2_O2_O2 = 4.1315;
    a_PdAgH075_T = 4.2134;
    a_PdAgH100_O1_O2_O2_O2 = 4.1848;
    a_PdAgH100_T = 4.3011;

    Ec_exp_PdAgH025_O1 = -3.4634;
    Ec_exp_PdAgH025_O2 = -3.4045;
    Ec_exp_PdAgH025_T = -3.4633;
    Ec_exp_PdAgH050_O1_O2 = -3.2946;
    Ec_exp_PdAgH050_O2_O2 = -3.2416;
    Ec_exp_PdAgH050_T = -3.2556;
    Ec_exp_PdAgH075_O1_O2_O2 = -3.1677;
    Ec_exp_PdAgH075_O2_O2_O2 = -3.1008;
    Ec_exp_PdAgH075_T = -3.1415;
    Ec_exp_PdAgH100_O1_O2_O2_O2 = -3.0465;
    Ec_exp_PdAgH100_T = -3.0388;

    a_PdAgH025_TE1 = 4.0016;
    a_PdAgH025_TE6 = 4.0016;
    a_PdAgH025_TE8 = 4.0016;

    a_PdAgH050_TE1_TE2 = 4.1239;
    a_PdAgH050_TE3_TE4 = 4.1239;
    a_PdAgH050_TE3_TE6 = 4.1239;
    a_PdAgH050_TE1_TE8 = 4.1387;
    a_PdAgH075_TE1_TE2_TE3 = 4.2027;
    a_PdAgH075_TE1_TE2_TE8 = 4.2203;
    a_PdAgH075_TE5_TE6_TE7 = 4.2134;
    a_PdAgH100_TE1_TE2_TE3_TE4 = 4.2874;
    a_PdAgH100_TE1_TE2_TE7_TE8 = 4.2983;
    a_PdAgH100_TE1_TE5_TE6_TE7 = 4.3011;

    Ec_exp_PdAgH025_TE1  = -3.4633;
    Ec_exp_PdAgH050_TE3_TE4 = -3.2556;
```

```
    Ec_exp_PdAgH050_TE1_TE8 = -3.2559;
    Ec_exp_PdAgH075_TE1_TE2_TE3 = -3.1126;
    Ec_exp_PdAgH075_TE1_TE2_TE8 = -3.1339;
    Ec_exp_PdAgH075_TE5_TE6_TE7 = -3.1415;
    Ec_exp_PdAgH100_TE1_TE2_TE3_TE4 = -3.0009;
    Ec_exp_PdAgH100_TE1_TE2_TE7_TE8 = -3.0065;
    Ec_exp_PdAgH100_TE1_TE5_TE6_TE7 = -3.0388;


    global rc_1
    global rc_2
    global rc_3

    rc_1 = 5.35;
    rc_2 = 5.35;
    rc_3 = 5.35;


    global x_Pd
    global x_Ag

    global S_Ag
    global S_Pd

    global x_PdH

% % %     Hijazi Yang Pd
    x_Pd = [2.054085    0.216817    8.414105    7.221224    0.999999
3.316887];
% % %
    x_Ag = [1.584768032776473   0.154164182657381   8.491335427574112...
   7.183185289310671   1.022270092304303   2.213230197455566];

% % %     re fit scaling only mixing rule potential
    S_Ag = 1.8319;
    S_Pd = 1.1063;

    x_PdH = [0.589510   1.104827    0.942490    2.145808    0.942201...
    0.740938    2.373944    1.702142    8.370790    62.343273...
    0.000100    1.187000    1.300000    3.474173];




% % %     Position Code For Tetrahedral H Atoms
% % % TE1   0.025   0.025   0.025
% % % TE2   0.075   0.025   0.025
% % % TE3   0.025   0.075   0.025
% % % TE4   0.025   0.025   0.075
% % % TE5   0.075   0.075   0.025
% % % TE6   0.025   0.075   0.075
% % % TE7   0.075   0.025   0.075
% % % TE8   0.075   0.075   0.075
```

```
% % %   Load Pd075Ag025H000 structure and find indices
    global ri_PdAgH000;
    global index_PdAgH000_Pd
    global index_PdAgH000_Ag
    load PdAgH000_sorted_6x6x6.data;
    ri_PdAgH000 = PdAgH000_sorted_6x6x6(:,:);
    ra_1 = [1 1 1 0];    % face center Pd
    ra_2 = [2 0 0 0];    % corner Ag
    index_PdAgH000_Pd = find(ismember(ri_PdAgH000,ra_1,'rows'));
    index_PdAgH000_Ag = find(ismember(ri_PdAgH000,ra_2,'rows'));


% % %   Load Pd075Ag025H025 (O1) structure and find indices
    global ri_PdAgH025_O1;
    global index_PdAgH025_O1_Pd1
    global index_PdAgH025_O1_Pd2
    global index_PdAgH025_O1_Pd3
    global index_PdAgH025_O1_Ag
    global index_PdAgH025_O1_H
    load PdAgH025_O1_sorted_6x6x6.dat;
    ri_PdAgH025_O1 = PdAgH025_O1_sorted_6x6x6(:,:);
    ra_1 = [1 1 1 0];    % face center Pd
    ra_2 = [1 1 0 1];    % face center Pd
    ra_3 = [1 0 1 1];    % face center Pd
    ra_4 = [2 0 0 0];    % corner Ag
    ra_5 = [3 1 1 1];    % body center H
    index_PdAgH025_O1_Pd1 = find(ismember(ri_PdAgH025_O1,ra_1,'rows'));
    index_PdAgH025_O1_Pd2 = find(ismember(ri_PdAgH025_O1,ra_2,'rows'));
    index_PdAgH025_O1_Pd3 = find(ismember(ri_PdAgH025_O1,ra_3,'rows'));
    index_PdAgH025_O1_Ag = find(ismember(ri_PdAgH025_O1,ra_4,'rows'));
    index_PdAgH025_O1_H = find(ismember(ri_PdAgH025_O1,ra_5,'rows'));


% % %   Load Pd075Ag025H025 (O2) structure and find indices
    global ri_PdAgH025_o2;
    global index_PdAgH025_o2_Pd1
    global index_PdAgH025_o2_Pd2
    global index_PdAgH025_o2_Pd3
    global index_PdAgH025_o2_Ag
    global index_PdAgH025_o2_H

    load PdAgH025_O2_sorted_6x6x6.dat;
    ri_PdAgH025_o2 = PdAgH025_O2_sorted_6x6x6(:,:);
    ra_1 = [1 1 1 0];    % face center Pd
    ra_2 = [2 0 0 0];    % corner Ag
    ra_3 = [3 1 0 0];    % edge H
    ra_4 = [1 1 0 1];    % face center Pd
    ra_5 = [1 0 1 1];    % face center Pd
    index_PdAgH025_o2_Pd1 = find(ismember(ri_PdAgH025_o2,ra_1,'rows'));
    index_PdAgH025_o2_Ag = find(ismember(ri_PdAgH025_o2,ra_2,'rows'));
    index_PdAgH025_o2_H = find(ismember(ri_PdAgH025_o2,ra_3,'rows'));
    index_PdAgH025_o2_Pd2 = find(ismember(ri_PdAgH025_o2,ra_4,'rows'));
    index_PdAgH025_o2_Pd3 = find(ismember(ri_PdAgH025_o2,ra_5,'rows'));
```

```
% % %   Load Pd075Ag025H050 (O1 O2) structure and find indices
    global ri_PdAgH050_O1_O2;
    global index_PdAgH050_O1_O2_Pd1
    global index_PdAgH050_O1_O2_Pd2
    global index_PdAgH050_O1_O2_Pd3
    global index_PdAgH050_O1_O2_Ag
    global index_PdAgH050_O1_O2_H_O1
    global index_PdAgH050_O1_O2_H_O2
    load PdAgH050_O1_O2_sorted_6x6x6.dat;
    ri_PdAgH050_O1_O2 = PdAgH050_O1_O2_sorted_6x6x6(:,:);
    ra_1 = [1 1 1 0];    % face center Pd
    ra_2 = [2 0 0 0];    % corner Ag
    ra_3 = [3 1 1 1];    % body center H
    ra_4 = [3 1 0 0];    % edge H
    ra_5 = [1 1 0 1];    % face center Pd
    ra_6 = [1 0 1 1];    % face center Pd
    index_PdAgH050_O1_O2_Pd1 = find(ismember(ri_PdAgH050_O1_O2,ra_1,'rows'));
    index_PdAgH050_O1_O2_Pd2 = find(ismember(ri_PdAgH050_O1_O2,ra_5,'rows'));
    index_PdAgH050_O1_O2_Pd3 = find(ismember(ri_PdAgH050_O1_O2,ra_6,'rows'));
    index_PdAgH050_O1_O2_Ag = find(ismember(ri_PdAgH050_O1_O2,ra_2,'rows'));
    index_PdAgH050_O1_O2_H_O1 =
find(ismember(ri_PdAgH050_O1_O2,ra_3,'rows'));
    index_PdAgH050_O1_O2_H_O2 =
find(ismember(ri_PdAgH050_O1_O2,ra_4,'rows'));


% % %   Load Pd075Ag025H050 (O2 O2) structure and find indices
    global ri_PdAgH050_O2_O2;
    global index_PdAgH050_O2_O2_Pd1
    global index_PdAgH050_O2_O2_Pd2
    global index_PdAgH050_O2_O2_Pd3
    global index_PdAgH050_O2_O2_Ag
    global index_PdAgH050_O2_O2_H1
    global index_PdAgH050_O2_O2_H3
    load PdAgH050_O2_O2_sorted_6x6x6.dat;
    ri_PdAgH050_O2_O2 = PdAgH050_O2_O2_sorted_6x6x6(:,:);
    ra_1 = [1 1 1 0];    % face center Pd
    ra_2 = [2 0 0 0];    % corner Ag
    ra_3 = [3 1 0 0];    % edge H
    ra_4 = [3 0 0 1];    % edge H
    ra_5 = [1 1 0 1];    % face center Pd
    ra_6 = [1 0 1 1];    % face center Pd
    index_PdAgH050_O2_O2_Pd1 = find(ismember(ri_PdAgH050_O2_O2,ra_1,'rows'));
    index_PdAgH050_O2_O2_Pd2 = find(ismember(ri_PdAgH050_O2_O2,ra_5,'rows'));
    index_PdAgH050_O2_O2_Pd3 = find(ismember(ri_PdAgH050_O2_O2,ra_6,'rows'));
    index_PdAgH050_O2_O2_Ag = find(ismember(ri_PdAgH050_O2_O2,ra_2,'rows'));
    index_PdAgH050_O2_O2_H1 = find(ismember(ri_PdAgH050_O2_O2,ra_3,'rows'));
    index_PdAgH050_O2_O2_H3 = find(ismember(ri_PdAgH050_O2_O2,ra_4,'rows'));


% % %   Load Pd075Ag025H025 (TE1) structure and find indices
    global ri_PdAgH025_TE1;
    global index_PdAgH025_TE1_Pd
    global index_PdAgH025_TE1_Ag
```

```
    global index_PdAgH025_TE1_H
    load PdAgH025_T_sorted_6x6x6_TE1.dat
    ri_PdAgH025_TE1 = PdAgH025_T_sorted_6x6x6_TE1(:,:);
    ra_1 = [1 1 1 0];    % face center Pd
    ra_2 = [2 0 0 0];    % corner Ag
    ra_3 = [3 .5 .5 .5];    % TE H
    index_PdAgH025_TE1_Pd = find(ismember(ri_PdAgH025_TE1,ra_1,'rows'));
    index_PdAgH025_TE1_Ag = find(ismember(ri_PdAgH025_TE1,ra_2,'rows'));
    index_PdAgH025_TE1_H = find(ismember(ri_PdAgH025_TE1,ra_3,'rows'));


% % %   Load Pd075Ag025H025 (TE6) structure and find indices
    global ri_PdAgH025_TE6;
    global index_PdAgH025_TE6_Pd
    global index_PdAgH025_TE6_Ag
    global index_PdAgH025_TE6_H
    load PdAgH025_T_sorted_6x6x6_TE6.dat
    ri_PdAgH025_TE6 = PdAgH025_T_sorted_6x6x6_TE6(:,:);
    ra_1 = [1 1 1 0];    % face center Pd
    ra_2 = [2 0 0 0];    % corner Ag
    ra_3 = [3 .5 1.5 1.5];    % TE H
    index_PdAgH025_TE6_Pd = find(ismember(ri_PdAgH025_TE6,ra_1,'rows'));
    index_PdAgH025_TE6_Ag = find(ismember(ri_PdAgH025_TE6,ra_2,'rows'));
    index_PdAgH025_TE6_H = find(ismember(ri_PdAgH025_TE6,ra_3,'rows'));


% % %   Load Pd075Ag025H025 (TE8) structure and find indices
    global ri_PdAgH025_TE8;
    global index_PdAgH025_TE8_Pd
    global index_PdAgH025_TE8_Ag
    global index_PdAgH025_TE8_H
    load PdAgH025_T_sorted_6x6x6_TE8.dat
    ri_PdAgH025_TE8 = PdAgH025_T_sorted_6x6x6_TE8(:,:);
    ra_1 = [1 1 1 0];    % face center Pd
    ra_2 = [2 0 0 0];    % corner Ag
    ra_3 = [3 1.5 1.5 1.5];    % TE H
    index_PdAgH025_TE8_Pd = find(ismember(ri_PdAgH025_TE8,ra_1,'rows'));
    index_PdAgH025_TE8_Ag = find(ismember(ri_PdAgH025_TE8,ra_2,'rows'));
    index_PdAgH025_TE8_H = find(ismember(ri_PdAgH025_TE8,ra_3,'rows'));


% % %   Load Pd075Ag025H050 (TE1 TE2) structure and find indices
    global ri_PdAgH050_TE1_TE2;
    global index_PdAgH050_TE1_TE2_Pd1
    global index_PdAgH050_TE1_TE2_Pd2
    global index_PdAgH050_TE1_TE2_Pd3
    global index_PdAgH050_TE1_TE2_Ag
    global index_PdAgH050_TE1_TE2_H_TE1
    global index_PdAgH050_TE1_TE2_H_TE2
    load PdAgH050_T_sorted_6x6x6_TE1_TE2.dat
    ri_PdAgH050_TE1_TE2 = PdAgH050_T_sorted_6x6x6_TE1_TE2(:,:);
    ra_1 = [1 1 1 0];    % face center Pd
    ra_2 = [1 1 0 1];    % face center Pd
    ra_3 = [1 0 1 1];    % face center Pd
    ra_4 = [2 0 0 0];    % corner Ag
    ra_5 = [3 .5 .5 .5];    % TE H
```

```
    ra_6 = [3 1.5 .5 .5];   % TE H
    index_PdAgH050_TE1_TE2_Pd1 =
find(ismember(ri_PdAgH050_TE1_TE2,ra_1,'rows'));
    index_PdAgH050_TE1_TE2_Pd2 =
find(ismember(ri_PdAgH050_TE1_TE2,ra_2,'rows'));
    index_PdAgH050_TE1_TE2_Pd3 =
find(ismember(ri_PdAgH050_TE1_TE2,ra_3,'rows'));
    index_PdAgH050_TE1_TE2_Ag =
find(ismember(ri_PdAgH050_TE1_TE2,ra_4,'rows'));
    index_PdAgH050_TE1_TE2_H_TE1 =
find(ismember(ri_PdAgH050_TE1_TE2,ra_5,'rows'));
    index_PdAgH050_TE1_TE2_H_TE2 =
find(ismember(ri_PdAgH050_TE1_TE2,ra_6,'rows'));


% % %   Load Pd075Ag025H050 (TE1 TE8) structure and find indices
    global ri_PdAgH050_TE1_TE8;
    global index_PdAgH050_TE1_TE8_Pd
    global index_PdAgH050_TE1_TE8_Ag
    global index_PdAgH050_TE1_TE8_H_TE1
    global index_PdAgH050_TE1_TE8_H_TE8
    load PdAgH050_T_sorted_6x6x6_TE1_TE8.dat
    ri_PdAgH050_TE1_TE8 = PdAgH050_T_sorted_6x6x6_TE1_TE8(:,:);
    ra_1 = [1 1 1 0];   % face center Pd
    ra_2 = [2 0 0 0];   % corner Ag
    ra_3 = [3 .5 .5 .5];   % TE H
    ra_4 = [3 1.5 1.5 1.5];   % TE H
    index_PdAgH050_TE1_TE8_Pd =
find(ismember(ri_PdAgH050_TE1_TE8,ra_1,'rows'));
    index_PdAgH050_TE1_TE8_Ag =
find(ismember(ri_PdAgH050_TE1_TE8,ra_2,'rows'));
    index_PdAgH050_TE1_TE8_H_TE1 =
find(ismember(ri_PdAgH050_TE1_TE8,ra_3,'rows'));
    index_PdAgH050_TE1_TE8_H_TE8 =
find(ismember(ri_PdAgH050_TE1_TE8,ra_4,'rows'));


% % %   Load Pd075Ag025H075 (O1 O2 O2) structure and find indices
    global ri_PdAgH075_O1_O2_O2;
    global index_PdAgH075_O1_O2_O2_Pd1
    global index_PdAgH075_O1_O2_O2_Pd2
    global index_PdAgH075_O1_O2_O2_Pd3
    global index_PdAgH075_O1_O2_O2_Ag
    global index_PdAgH075_O1_O2_O2_H1
    global index_PdAgH075_O1_O2_O2_H2
    global index_PdAgH075_O1_O2_O2_H3
    load PdAgH075_O1_O2_O2_sorted_6x6x6.dat;
    ri_PdAgH075_O1_O2_O2 = PdAgH075_O1_O2_O2_sorted_6x6x6(:,:);
    ra_1 = [1 1 1 0];   % face center Pd
    ra_5 = [1 1 0 1];   % face center Pd
    ra_6 = [1 0 1 1];   % face center Pd
    ra_2 = [2 0 0 0];   % corner Ag
    ra_3 = [3 1 1 1];   % body center H
    ra_4 = [3 1 0 0];   % edge H
    ra_7 = [3 0 0 1];   % edge H
```

```
    index_PdAgH075_O1_O2_O2_Pd1 =
find(ismember(ri_PdAgH075_O1_O2_O2,ra_1,'rows'));
    index_PdAgH075_O1_O2_O2_Pd2 =
find(ismember(ri_PdAgH075_O1_O2_O2,ra_5,'rows'));
    index_PdAgH075_O1_O2_O2_Pd3 =
find(ismember(ri_PdAgH075_O1_O2_O2,ra_6,'rows'));
    index_PdAgH075_O1_O2_O2_Ag =
find(ismember(ri_PdAgH075_O1_O2_O2,ra_2,'rows'));
    index_PdAgH075_O1_O2_O2_H1 =
find(ismember(ri_PdAgH075_O1_O2_O2,ra_3,'rows'));
    index_PdAgH075_O1_O2_O2_H2 =
find(ismember(ri_PdAgH075_O1_O2_O2,ra_4,'rows'));
    index_PdAgH075_O1_O2_O2_H3 =
find(ismember(ri_PdAgH075_O1_O2_O2,ra_7,'rows'));


% % %   Load Pd075Ag025H075 (O2 O2 O2) structure and find indices
    global ri_PdAgH075_O2_O2_O2;
    global index_PdAgH075_O2_O2_O2_Pd1
    global index_PdAgH075_O2_O2_O2_Pd2
    global index_PdAgH075_O2_O2_O2_Pd3
    global index_PdAgH075_O2_O2_O2_Ag
    global index_PdAgH075_O2_O2_O2_H1
    global index_PdAgH075_O2_O2_O2_H2
    global index_PdAgH075_O2_O2_O2_H3
    load PdAgH075_O2_O2_O2_sorted_6x6x6.dat;
    ri_PdAgH075_O2_O2_O2 = PdAgH075_O2_O2_O2_sorted_6x6x6(:,:);
    ra_1 = [1 1 1 0];    % face center Pd
    ra_2 = [1 1 0 1];    % face center Pd
    ra_3 = [1 0 1 1];    % face center Pd
    ra_4 = [2 0 0 0];    % corner Ag
    ra_5 = [3 1 0 0];    % edge H
    ra_6 = [3 0 1 0];    % edge H
    ra_7 = [3 0 0 1];    % edge H
    index_PdAgH075_O2_O2_O2_Pd1 =
find(ismember(ri_PdAgH075_O2_O2_O2,ra_1,'rows'));
    index_PdAgH075_O2_O2_O2_Pd2 =
find(ismember(ri_PdAgH075_O2_O2_O2,ra_2,'rows'));
    index_PdAgH075_O2_O2_O2_Pd3 =
find(ismember(ri_PdAgH075_O2_O2_O2,ra_3,'rows'));
    index_PdAgH075_O2_O2_O2_Ag =
find(ismember(ri_PdAgH075_O2_O2_O2,ra_4,'rows'));
    index_PdAgH075_O2_O2_O2_H1 =
find(ismember(ri_PdAgH075_O2_O2_O2,ra_5,'rows'));
    index_PdAgH075_O2_O2_O2_H2 =
find(ismember(ri_PdAgH075_O2_O2_O2,ra_6,'rows'));
    index_PdAgH075_O2_O2_O2_H3 =
find(ismember(ri_PdAgH075_O2_O2_O2,ra_7,'rows'));


% % %   Load structure Pd075Ag025H075 (TE5 TE6 TE7) and find indices
    global ri_PdAgH075_T;
    global index_PdAgH075_T_Pd
    global index_PdAgH075_T_Ag
    global index_PdAgH075_T_H
    load PdAgH075_T_sorted_6x6x6.dat;
```

```
    ri_PdAgH075_T = PdAgH075_T_sorted_6x6x6(:,:);
    ra_1 = [1 1 1 0];    % face center Pd
    ra_2 = [2 0 0 0];    % corner Ag
    ra_3 = [3 .5 .5 .5];    % TE H
%     ra_4 = [3 1.5 .5 .5];    % TE H
    index_PdAgH075_T_Pd = find(ismember(ri_PdAgH075_T,ra_1,'rows'));
    index_PdAgH075_T_Ag = find(ismember(ri_PdAgH075_T,ra_2,'rows'));
    index_PdAgH075_T_H = find(ismember(ri_PdAgH075_T,ra_3,'rows'));
%     index_PdAgH075_T_H = find(ismember(ri_PdAgH075_T,ra_4,'rows'));


% % %   Load Pd075Ag025H100 (O1 O2 O2 O2) structure and find indices
    global ri_PdAgH100_O1_O2_O2_O2;
    global index_PdAgH100_O1_O2_O2_O2_Pd1
    global index_PdAgH100_O1_O2_O2_O2_Pd2
    global index_PdAgH100_O1_O2_O2_O2_Pd3
    global index_PdAgH100_O1_O2_O2_O2_Ag
    global index_PdAgH100_O1_O2_O2_O2_H1
    global index_PdAgH100_O1_O2_O2_O2_H2
    global index_PdAgH100_O1_O2_O2_O2_H3
    global index_PdAgH100_O1_O2_O2_O2_H4
    load PdAgH100_O1_O2_O2_O2_sorted_6x6x6.dat;
    ri_PdAgH100_O1_O2_O2_O2 = PdAgH100_O1_O2_O2_O2_sorted_6x6x6(:,:);
    ra_1 = [1 1 1 0];    % face center Pd
    ra_5 = [1 1 0 1];    % face center Pd
    ra_6 = [1 0 1 1];    % face center Pd
    ra_2 = [2 0 0 0];    % corner Ag
    ra_3 = [3 1 1 1];    % body center H
    ra_4 = [3 1 0 0];    % edge H
    ra_7 = [3 0 1 0];    % edge H
    ra_8 = [3 0 0 1];    % edge H
    index_PdAgH100_O1_O2_O2_O2_Pd1 =
find(ismember(ri_PdAgH100_O1_O2_O2_O2,ra_1,'rows'));
    index_PdAgH100_O1_O2_O2_O2_Pd2 =
find(ismember(ri_PdAgH100_O1_O2_O2_O2,ra_5,'rows'));
    index_PdAgH100_O1_O2_O2_O2_Pd3 =
find(ismember(ri_PdAgH100_O1_O2_O2_O2,ra_6,'rows'));
    index_PdAgH100_O1_O2_O2_O2_Ag =
find(ismember(ri_PdAgH100_O1_O2_O2_O2,ra_2,'rows'));
    index_PdAgH100_O1_O2_O2_O2_H1 =
find(ismember(ri_PdAgH100_O1_O2_O2_O2,ra_3,'rows'));
    index_PdAgH100_O1_O2_O2_O2_H2 =
find(ismember(ri_PdAgH100_O1_O2_O2_O2,ra_4,'rows'));
    index_PdAgH100_O1_O2_O2_O2_H3 =
find(ismember(ri_PdAgH100_O1_O2_O2_O2,ra_7,'rows'));
    index_PdAgH100_O1_O2_O2_O2_H4 =
find(ismember(ri_PdAgH100_O1_O2_O2_O2,ra_8,'rows'));


% % %   Load Pd075Ag025H100 (TE1 TE5 TE6 TE7) structure and find indices
    global ri_PdAgH100_T;
    global index_PdAgH100_T_Pd
    global index_PdAgH100_T_Ag
    global index_PdAgH100_T_H
    load PdAgH100_T_sorted_6x6x6.dat;
    ri_PdAgH100_T = PdAgH100_T_sorted_6x6x6(:,:);
```

```
    ra_1 = [1 1 1 0];    % face center Pd
    ra_2 = [2 0 0 0];    % corner Ag
    ra_3 = [3 .5 .5 .5];    % TE H
%     ra_4 = [3 1.5 .5 .5];    % TE H
    index_PdAgH100_T_Pd = find(ismember(ri_PdAgH100_T,ra_1,'rows'));
    index_PdAgH100_T_Ag = find(ismember(ri_PdAgH100_T,ra_2,'rows'));
    index_PdAgH100_T_H = find(ismember(ri_PdAgH100_T,ra_3,'rows'));
%     index_PdAgH100_T_H = find(ismember(ri_PdAgH100_T,ra_4,'rows'));


% % %   Load Pd075Ag025H050 (TE3 TE4) structure and find indices
    global ri_PdAgH050_TE3_TE4
    global index_PdAgH050_TE3_TE4_Pd
    global index_PdAgH050_TE3_TE4_Ag
    global index_PdAgH050_TE3_TE4_H_TE3
    global index_PdAgH050_TE3_TE4_H_TE4
    load PdAgH050_T_sorted_6x6x6_TE3_TE4.dat;
    ri_PdAgH050_TE3_TE4 = PdAgH050_T_sorted_6x6x6_TE3_TE4(:,:);
    ra_1 = [1 1 1 0];    % face center Pd
    ra_2 = [2 0 0 0];    % corner Ag
    ra_3 = [3 .5 1.5 .5];    % TE H
    ra_4 = [3 .5 .5 1.5];    % TE H
    index_PdAgH050_TE3_TE4_Pd =
find(ismember(ri_PdAgH050_TE3_TE4,ra_1,'rows'));
    index_PdAgH050_TE3_TE4_Ag =
find(ismember(ri_PdAgH050_TE3_TE4,ra_2,'rows'));
    index_PdAgH050_TE3_TE4_H_TE3 =
find(ismember(ri_PdAgH050_TE3_TE4,ra_3,'rows'));
    index_PdAgH050_TE3_TE4_H_TE4 =
find(ismember(ri_PdAgH050_TE3_TE4,ra_4,'rows'));


% % %   Load Pd075Ag025H050 (TE3 TE6) structure and find indices
    global ri_PdAgH050_TE3_TE6
    global index_PdAgH050_TE3_TE6_Pd
    global index_PdAgH050_TE3_TE6_Ag
    global index_PdAgH050_TE3_TE6_H_TE3
    global index_PdAgH050_TE3_TE6_H_TE6
    load PdAgH050_T_sorted_6x6x6_TE3_TE6.dat;
    ri_PdAgH050_TE3_TE6 = PdAgH050_T_sorted_6x6x6_TE3_TE6(:,:);
    ra_1 = [1 1 1 0];    % face center Pd
    ra_2 = [2 0 0 0];    % corner Ag
    ra_3 = [3 .5 1.5 .5];    % TE H
    ra_4 = [3 .5 1.5 1.5];    % TE H
    index_PdAgH050_TE3_TE6_Pd =
find(ismember(ri_PdAgH050_TE3_TE6,ra_1,'rows'));
    index_PdAgH050_TE3_TE6_Ag =
find(ismember(ri_PdAgH050_TE3_TE6,ra_2,'rows'));
    index_PdAgH050_TE3_TE6_H_TE3 =
find(ismember(ri_PdAgH050_TE3_TE6,ra_3,'rows'));
    index_PdAgH050_TE3_TE6_H_TE6 =
find(ismember(ri_PdAgH050_TE3_TE6,ra_4,'rows'));


% % %   Load Pd075Ag025H075 (TE1 TE2 TE3) structure and find indices
    global ri_PdAgH075_TE1_TE2_TE3
```

```
    global index_PdAgH075_TE1_TE2_TE3_Pd1
    global index_PdAgH075_TE1_TE2_TE3_Pd2
    global index_PdAgH075_TE1_TE2_TE3_Pd3
    global index_PdAgH075_TE1_TE2_TE3_Ag
    global index_PdAgH075_TE1_TE2_TE3_H_TE1
    global index_PdAgH075_TE1_TE2_TE3_H_TE2
    global index_PdAgH075_TE1_TE2_TE3_H_TE3
    load PdAgH075_T_sorted_6x6x6_TE1_TE2_TE3.dat;
    ri_PdAgH075_TE1_TE2_TE3 = PdAgH075_T_sorted_6x6x6_TE1_TE2_TE3(:,:);
    ra_1 = [1 1 1 0];    % face center Pd
    ra_2 = [1 1 0 1];    % face center Pd
    ra_3 = [1 0 1 1];    % face center Pd
    ra_4 = [2 0 0 0];    % corner Ag
    ra_5 = [3 .5 .5 .5];    % TE H
    ra_6 = [3 1.5 .5 .5];    % TE H
    ra_7 = [3 .5 1.5 .5];    % TE H
    index_PdAgH075_TE1_TE2_TE3_Pd1 =
find(ismember(ri_PdAgH075_TE1_TE2_TE3,ra_1,'rows'));
    index_PdAgH075_TE1_TE2_TE3_Pd2 =
find(ismember(ri_PdAgH075_TE1_TE2_TE3,ra_2,'rows'));
    index_PdAgH075_TE1_TE2_TE3_Pd3 =
find(ismember(ri_PdAgH075_TE1_TE2_TE3,ra_3,'rows'));
    index_PdAgH075_TE1_TE2_TE3_Ag =
find(ismember(ri_PdAgH075_TE1_TE2_TE3,ra_4,'rows'));
    index_PdAgH075_TE1_TE2_TE3_H_TE1 =
find(ismember(ri_PdAgH075_TE1_TE2_TE3,ra_5,'rows'));
    index_PdAgH075_TE1_TE2_TE3_H_TE2 =
find(ismember(ri_PdAgH075_TE1_TE2_TE3,ra_6,'rows'));
    index_PdAgH075_TE1_TE2_TE3_H_TE3 =
find(ismember(ri_PdAgH075_TE1_TE2_TE3,ra_7,'rows'));


% % %   Load Pd075Ag025H075 (TE1 TE2 TE8) structure and find indices
    global ri_PdAgH075_TE1_TE2_TE8
    global index_PdAgH075_TE1_TE2_TE8_Pd1
    global index_PdAgH075_TE1_TE2_TE8_Pd2
    global index_PdAgH075_TE1_TE2_TE8_Pd3
    global index_PdAgH075_TE1_TE2_TE8_Ag
    global index_PdAgH075_TE1_TE2_TE8_H_TE1
    global index_PdAgH075_TE1_TE2_TE8_H_TE2
    global index_PdAgH075_TE1_TE2_TE8_H_TE8
    load PdAgH075_T_sorted_6x6x6_TE1_TE2_TE8.dat;
    ri_PdAgH075_TE1_TE2_TE8 = PdAgH075_T_sorted_6x6x6_TE1_TE2_TE8(:,:);
    ra_1 = [1 1 1 0];    % face center Pd
    ra_2 = [1 1 0 1];    % face center Pd
    ra_3 = [1 0 1 1];    % face center Pd
    ra_4 = [2 0 0 0];    % corner Ag
    ra_5 = [3 .5 .5 .5];    % TE H
    ra_6 = [3 1.5 .5 .5];    % TE H
    ra_7 = [3 1.5 1.5 1.5];    % TE H
    index_PdAgH075_TE1_TE2_TE8_Pd1 =
find(ismember(ri_PdAgH075_TE1_TE2_TE8,ra_1,'rows'));
    index_PdAgH075_TE1_TE2_TE8_Pd2 =
find(ismember(ri_PdAgH075_TE1_TE2_TE8,ra_2,'rows'));
    index_PdAgH075_TE1_TE2_TE8_Pd3 =
find(ismember(ri_PdAgH075_TE1_TE2_TE8,ra_3,'rows'));
```

```
    index_PdAgH075_TE1_TE2_TE8_Ag =
find(ismember(ri_PdAgH075_TE1_TE2_TE8,ra_4,'rows'));
    index_PdAgH075_TE1_TE2_TE8_H_TE1 =
find(ismember(ri_PdAgH075_TE1_TE2_TE8,ra_5,'rows'));
    index_PdAgH075_TE1_TE2_TE8_H_TE2 =
find(ismember(ri_PdAgH075_TE1_TE2_TE8,ra_6,'rows'));
    index_PdAgH075_TE1_TE2_TE8_H_TE8 =
find(ismember(ri_PdAgH075_TE1_TE2_TE8,ra_7,'rows'));


% % %   Load Pd075Ag025H100 (TE1 TE2 TE3 TE4) structure and find indices
    global ri_PdAgH100_TE1_TE2_TE3_TE4
    global index_PdAgH100_TE1_TE2_TE3_TE4_Pd1
    global index_PdAgH100_TE1_TE2_TE3_TE4_Pd2
    global index_PdAgH100_TE1_TE2_TE3_TE4_Pd3
    global index_PdAgH100_TE1_TE2_TE3_TE4_Ag
    global index_PdAgH100_TE1_TE2_TE3_TE4_H_TE1
    global index_PdAgH100_TE1_TE2_TE3_TE4_H_TE2
    global index_PdAgH100_TE1_TE2_TE3_TE4_H_TE3
    global index_PdAgH100_TE1_TE2_TE3_TE4_H_TE4
    load PdAgH100_T_sorted_6x6x6_TE1_TE2_TE3_TE4.dat;
    ri_PdAgH100_TE1_TE2_TE3_TE4 =
PdAgH100_T_sorted_6x6x6_TE1_TE2_TE3_TE4(:,:);
    ra_1 = [1 1 1 0];   % face center Pd
    ra_2 = [1 1 0 1];   % face center Pd
    ra_3 = [1 0 1 1];   % face center Pd
    ra_4 = [2 0 0 0];   % corner Ag
    ra_5 = [3 .5 .5 .5];   % TE H
    ra_6 = [3 1.5 .5 .5];   % TE H
    ra_7 = [3 .5 1.5 .5];   % TE H
    ra_8 = [3 .5 .5 1.5];   % TE H
    index_PdAgH100_TE1_TE2_TE3_TE4_Pd1 =
find(ismember(ri_PdAgH100_TE1_TE2_TE3_TE4,ra_1,'rows'));
    index_PdAgH100_TE1_TE2_TE3_TE4_Pd2 =
find(ismember(ri_PdAgH100_TE1_TE2_TE3_TE4,ra_2,'rows'));
    index_PdAgH100_TE1_TE2_TE3_TE4_Pd3 =
find(ismember(ri_PdAgH100_TE1_TE2_TE3_TE4,ra_3,'rows'));
    index_PdAgH100_TE1_TE2_TE3_TE4_Ag =
find(ismember(ri_PdAgH100_TE1_TE2_TE3_TE4,ra_4,'rows'));
    index_PdAgH100_TE1_TE2_TE3_TE4_H_TE1 =
find(ismember(ri_PdAgH100_TE1_TE2_TE3_TE4,ra_5,'rows'));
    index_PdAgH100_TE1_TE2_TE3_TE4_H_TE2 =
find(ismember(ri_PdAgH100_TE1_TE2_TE3_TE4,ra_6,'rows'));
    index_PdAgH100_TE1_TE2_TE3_TE4_H_TE3 =
find(ismember(ri_PdAgH100_TE1_TE2_TE3_TE4,ra_7,'rows'));
    index_PdAgH100_TE1_TE2_TE3_TE4_H_TE4 =
find(ismember(ri_PdAgH100_TE1_TE2_TE3_TE4,ra_8,'rows'));


% % %   Load Pd075Ag025H100 (TE1 TE2 TE7 TE8) structure and find indices
    global ri_PdAgH100_TE1_TE2_TE7_TE8
    global index_PdAgH100_TE1_TE2_TE7_TE8_Pd
    global index_PdAgH100_TE1_TE2_TE7_TE8_Ag
    global index_PdAgH100_TE1_TE2_TE7_TE8_H_TE1
    global index_PdAgH100_TE1_TE2_TE7_TE8_H_TE2
    global index_PdAgH100_TE1_TE2_TE7_TE8_H_TE7
```

```
    global index_PdAgH100_TE1_TE2_TE7_TE8_H_TE8
    load PdAgH100_T_sorted_6x6x6_TE1_TE2_TE7_TE8.dat;
    ri_PdAgH100_TE1_TE2_TE7_TE8 =
PdAgH100_T_sorted_6x6x6_TE1_TE2_TE7_TE8(:,:);
    ra_1 = [1 1 1 0];   % face center Pd
    ra_2 = [2 0 0 0];   % corner Ag
    ra_3 = [3 .5 .5 .5];   % TE H
    ra_4 = [3 1.5 .5 .5];   % TE H
    ra_5 = [3 1.5 .5 1.5];   % TE H
    ra_6 = [3 1.5 1.5 1.5];   % TE H
    index_PdAgH100_TE1_TE2_TE7_TE8_Pd =
find(ismember(ri_PdAgH100_TE1_TE2_TE7_TE8,ra_1,'rows'));
    index_PdAgH100_TE1_TE2_TE7_TE8_Ag =
find(ismember(ri_PdAgH100_TE1_TE2_TE7_TE8,ra_2,'rows'));
    index_PdAgH100_TE1_TE2_TE7_TE8_H_TE1 =
find(ismember(ri_PdAgH100_TE1_TE2_TE7_TE8,ra_3,'rows'));
    index_PdAgH100_TE1_TE2_TE7_TE8_H_TE2 =
find(ismember(ri_PdAgH100_TE1_TE2_TE7_TE8,ra_4,'rows'));
    index_PdAgH100_TE1_TE2_TE7_TE8_H_TE7 =
find(ismember(ri_PdAgH100_TE1_TE2_TE7_TE8,ra_5,'rows'));
    index_PdAgH100_TE1_TE2_TE7_TE8_H_TE8 =
find(ismember(ri_PdAgH100_TE1_TE2_TE7_TE8,ra_6,'rows'));



% % % Fitting Constraints


% %  Used in Thesis
% % % % %    f = 0.4417     x = 1.476745339175117   1.967649395899897
1.741864506174084   1.850016663067700
    Xl_AgH = [.1        1         1          1.85];
    X0_AgH = [.1        1         1          3];
    Xu_AgH = [10        10        10        3];
x = [1.476745339175117   1.967649395899897   1.741864506174084
1.850016663067700 ]



    tic  % start timer

    options.MaxFunEvals = 70000;
    options.TolFun = 1.00e-7;
    options.TolX = 1.00e-7;

%     %%%% Perform Constrained Optimization
%     [x,fval] =
fmincon(@objfun_18_07_13,X0_AgH,[],[],[],[],Xl_AgH,Xu_AgH,[],options)

%     %%%% Perform Unconstrained Optimization
% % % %     [x,fval] = fminunc(@objfun_18_03_22,X0_AgH,options)

%     %%%% Perform Genetic Algorithm Optimization
%     rng('default')
```

```
%     options =
optimoptions(@ga,'Generations',500,'PopulationSize',100,'HybridFcn',@fmincon)
;
%     [x, fval] = ga(@objfun_18_07_13,4,[],[],[],[], Xl_AgH ,Xu_AgH
,[],options)

    toc  % stop timer


    Check_Fit_18_07_13(x,false,false)


    objfun_18_07_13(x);
    format long
    x

end
```

objfunc.m

```
function f = objfun(x,weights)

    global Ec_exp_PdAgH025_O1;
    global Ec_exp_PdAgH025_O2;
    global Ec_exp_PdAgH050_O1_O2;
    global Ec_exp_PdAgH050_O2_O2;
    global Ec_exp_PdAgH075_O1_O2_O2;
    global Ec_exp_PdAgH075_O2_O2_O2;
    global Ec_exp_PdAgH100_O1_O2_O2_O2;

    global Ec_exp_PdAgH025_TE1
    global Ec_exp_PdAgH050_TE3_TE4
    global Ec_exp_PdAgH050_TE1_TE8
    global Ec_exp_PdAgH075_TE1_TE2_TE3
    global Ec_exp_PdAgH075_TE1_TE2_TE8
    global Ec_exp_PdAgH075_TE5_TE6_TE7
    global Ec_exp_PdAgH100_TE1_TE2_TE3_TE4
    global Ec_exp_PdAgH100_TE1_TE2_TE7_TE8
    global Ec_exp_PdAgH100_TE1_TE5_TE6_TE7


    global x_Pd
    global x_Ag

    global S_Ag
    global S_Pd

    global x_PdH
```

```
global rc_1
global rc_2
global rc_3


% Pd Experimental data
[aFcce_Pd,EcFcce_Pd,C11e_Pd,C12e_Pd,C44e_Pd,Bme_Pd,Eve_Pd,aBcce_Pd,...
    EcBcce_Pd] = parameters('Pd');

re = aFcce_Pd/sqrt(2);
Fe = EcFcce_Pd-Eve_Pd;
N  = 4;
V  = aFcce_Pd^3/N;
fe = EcFcce_Pd/V;

fe_S = S_Pd*EcFcce_Pd/V;

% Pd Fitting Parameters
Xi   = x_Pd(1);
phie = x_Pd(2);
S    = x_Pd(3);
B    = x_Pd(4);
n    = x_Pd(5);
rhoe = x_Pd(6);

%  Rho PdPd fitting parameters
X0_PdPd = [fe,Xi,re];
X0_PdPd_S = [fe_S,Xi,re];
%  F PdPd fitting parameters
X1_PdPd = [Fe,rhoe,n];
%  Phi PdPd fitting parametters
X2_PdPd = [phie,S,B,re];


% Ag Experimental data
[aFcce_Ag,EcFcce_Ag,C11e_Ag,C12e_Ag,C44e_Ag,Bme_Ag,Eve_Ag,aBcce_Ag,...
    EcBcce_Ag] = parameters('Ag');

re = aFcce_Ag/sqrt(2);
Fe = EcFcce_Ag-Eve_Ag;
N  = 4;
V  = aFcce_Ag^3/N;
fe = EcFcce_Ag/V;

fe_S = S_Ag*EcFcce_Ag/V;


% Ag Fitting Parameters
Xi   = x_Ag(1);
phie = x_Ag(2);
S    = x_Ag(3);
B    = x_Ag(4);
```

```
    n    = x_Ag(5);
    rhoe = x_Ag(6);

    %  Rho PdPd fitting parameters
    X0_AgAg = [fe,Xi,re];
    X0_AgAg_S = [fe_S,Xi,re];
    %  F PdPd fitting parameters
    X1_AgAg = [Fe,rhoe,n];
    %  Phi PdPd fitting parametters
    X2_AgAg = [phie,S,B,re];

    X5_PdAg = [X0_PdPd_S X0_AgAg_S X2_PdPd X2_AgAg];



    % Phi_HH
    DHH =  x_PdH(1);
    aHH =  x_PdH(2);
    bHH =  x_PdH(3);
    % f_HH
    CH = x_PdH(4);
    DH = x_PdH(5);
    % Phi_PdH
    DPdH = x_PdH(6);
    aPdH = x_PdH(7);
    bPdH = x_PdH(8);
    % F_H
    aH = x_PdH(9);
    bH = x_PdH(10);
    cH = x_PdH(11);
    dH = x_PdH(12);
    %
    r0PdH = x_PdH(13);
    r0HH =  x_PdH(14);

    % f_HH
    X0_HH  = [CH,DH];

%     % rho0H calculation
%     ri    = ri_PdH100_OC;
%     index = index_PdH100_OC_1;
%     aFcce_H = 3.38;
%     rho0H = Rho(index,1,ri,aFcce_H,rc_2,@f_HH,X0_HH)
    rho0H = 4.903820;

    % F HH fitting parameters
    X1_HH  = [aH,bH,cH,dH,rho0H];
    % Phi HH fitting parametters
    X2_HH  = [DHH,aHH,bHH,r0HH];
    %
    X22_HH = [X2_HH,X0_HH,X1_HH];
    % Phi PdH fitting parametters
    X4_PdH = [DPdH,aPdH,bPdH,r0PdH];
```

```
    % Phi_AgH
    DAgH = x(1);
    aAgH = x(2);
    bAgH = x(3);
    r0AgH = x(4);

    % Phi AgH fitting parametters
    X4_AgH = [DAgH,aAgH,bAgH,r0AgH];


%*************************************************************************
% For Pd075Ag025H025_O1
%*************************************************************************
    global ri_PdAgH025_O1;
    global index_PdAgH025_O1_Pd1
    global index_PdAgH025_O1_Pd2
    global index_PdAgH025_O1_Pd3
    global index_PdAgH025_O1_Ag
    global index_PdAgH025_O1_H

    ri = ri_PdAgH025_O1;
    index_Pd1 = index_PdAgH025_O1_Pd1;
    index_Pd2 = index_PdAgH025_O1_Pd2;
    index_Pd3 = index_PdAgH025_O1_Pd3;
    index_Ag = index_PdAgH025_O1_Ag;
    index_H  = index_PdAgH025_O1_H;

    global a_PdAgH025_o1;
    a = a_PdAgH025_o1;

    XX = 1;
    YY = 1;
    ZZ = 1;

    xx = 0.25;
    yy = 0.25;
    zz = 0.25;

    % Pd Central
    Ec_Pd_12 =    Ec3(ri,a,index_Pd1,1,XX,...
                    @f_PdPd,X0_PdPd,...
                    @F_Pd,X1_PdPd,...
                    @phi_PdPd,X2_PdPd,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_PdH,rc_1);
    % Pd Central
    Ec_Pd_22 =    Ec3(ri,a,index_Pd2,1,XX,...
                    @f_PdPd,X0_PdPd,...
                    @F_Pd,X1_PdPd,...
                    @phi_PdPd,X2_PdPd,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
```

```
                    @phi_PdH,X4_PdH,rc_1);
    % Pd Central
    Ec_Pd_32 =    Ec3(ri,a,index_Pd3,1,XX,...
                    @f_PdPd,X0_PdPd,...
                    @F_Pd,X1_PdPd,...
                    @phi_PdPd,X2_PdPd,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_PdH,rc_1);

     % Ag Central
    Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
                   @f_PdPd,X0_AgAg,...
                    @F_Pd,X1_AgAg,...
                   @phi_PdPd,X2_AgAg,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_AgH,rc_1);

     % H Central
    Ec_H_12 =    Ec3(ri,a,index_H,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

    Ec_Pd075Ag025H025_O1   = ( xx * Ec_Pd_12 + xx * Ec_Pd_22 + xx * Ec_Pd_32
+ yy * Ec_Ag_12 + zz * Ec_H_12)/(3*xx + yy + zz);

%************************************************************************
% For Pd075Ag025H025_O2
%************************************************************************
    global ri_PdAgH025_o2;
    global index_PdAgH025_o2_Pd1
    global index_PdAgH025_o2_Pd2
    global index_PdAgH025_o2_Pd3
    global index_PdAgH025_o2_Ag
    global index_PdAgH025_o2_H

    ri = ri_PdAgH025_o2;
    index_Pd1 = index_PdAgH025_o2_Pd1;
    index_Pd2 = index_PdAgH025_o2_Pd2;
    index_Pd3 = index_PdAgH025_o2_Pd3;
    index_Ag = index_PdAgH025_o2_Ag;
    index_H  = index_PdAgH025_o2_H;

    global a_PdAgH025_o2;
    a = a_PdAgH025_o2;
```

```
XX = 1;
YY = 1;
ZZ = 1;

xx = 0.25;
yy = 0.25;
zz = 0.25;

% Pd Central
Ec_Pd_12 =    Ec3(ri,a,index_Pd1,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);

% Pd Central
Ec_Pd_22 =    Ec3(ri,a,index_Pd2,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
% Pd Central
Ec_Pd_32 =    Ec3(ri,a,index_Pd3,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);

 % Ag Central
Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
                @f_PdPd,X0_AgAg,...
                 @F_Pd,X1_AgAg,...
                @phi_PdPd,X2_AgAg,rc_1,...
                1,XX,@f_PdPd,X0_PdPd,...
                 @phi_PdAg,X5_PdAg,rc_1,...
                 3,ZZ,@f_HH,X0_HH,...
                 @phi_PdH,X4_AgH,rc_1);

 % H Central
Ec_H_12 =    Ec3(ri,a,index_H,3,ZZ,...
                @f_HH,X0_HH,...
                 @F_H,X22_HH,...
                @phi_HH,X22_HH,rc_1,...
                1,XX,@f_PdPd,X0_PdPd,...
                 @phi_PdH,X4_PdH,rc_1,...
                 2,YY,@f_PdPd,X0_AgAg,...
                 @phi_PdH,X4_AgH,rc_1);
```

```
    Ec_Pd075Ag025H025_O2   = ( xx * Ec_Pd_12 + xx * Ec_Pd_22 + xx * Ec_Pd_32
+ yy * Ec_Ag_12 + zz * Ec_H_12)/(3*xx + yy + zz);

%************************************************************************
% For Pd075Ag025H025_TE1
%************************************************************************


    global ri_PdAgH025_TE1;
    global index_PdAgH025_TE1_Pd
    global index_PdAgH025_TE1_Ag
    global index_PdAgH025_TE1_H

    ri = ri_PdAgH025_TE1;
    index_Pd = index_PdAgH025_TE1_Pd;
    index_Ag = index_PdAgH025_TE1_Ag;
    index_H  = index_PdAgH025_TE1_H;

    global a_PdAgH025_TE1
    a = a_PdAgH025_TE1;

    XX = 1;
    YY = 1;
    ZZ = 1;

    xx = 0.75;
    yy = 0.25;
    zz = 0.25;

    % Pd Central
    Ec_Pd_12 =    Ec3(ri,a,index_Pd,1,XX,...
                    @f_PdPd,X0_PdPd,...
                    @F_Pd,X1_PdPd,...
                    @phi_PdPd,X2_PdPd,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_PdH,rc_1);
    % Ag Central
    Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
                   @f_PdPd,X0_AgAg,...
                    @F_Pd,X1_AgAg,...
                   @phi_PdPd,X2_AgAg,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_AgH,rc_1);

     % H Central
    Ec_H_12 =    Ec3(ri,a,index_H,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
```

```
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

    Ec_Pd075Ag025H025_TE1   = ( xx * Ec_Pd_12  + yy * Ec_Ag_12 + zz *
Ec_H_12)/(xx + yy + zz);


%************************************************************************
% For Pd075Ag025H050_O1_O2
%************************************************************************
    global ri_PdAgH050_O1_O2;
    global index_PdAgH050_O1_O2_Pd1
    global index_PdAgH050_O1_O2_Pd2
    global index_PdAgH050_O1_O2_Pd3
    global index_PdAgH050_O1_O2_Ag
    global index_PdAgH050_O1_O2_H_O1
    global index_PdAgH050_O1_O2_H_O2

    ri = ri_PdAgH050_O1_O2;
    index_Pd1 = index_PdAgH050_O1_O2_Pd1;
    index_Pd2 = index_PdAgH050_O1_O2_Pd2;
    index_Pd3 = index_PdAgH050_O1_O2_Pd3;
    index_Ag = index_PdAgH050_O1_O2_Ag;
    index_H1  = index_PdAgH050_O1_O2_H_O1;
    index_H2  = index_PdAgH050_O1_O2_H_O2;

    global a_PdAgH050_O1_O2;
    a = a_PdAgH050_O1_O2;



    XX = 1;
    YY = 1;
    ZZ = 1;

    xx = 0.25;
    yy = 0.25;
    zz = 0.25;

    % Pd Central
    Ec_Pd_12 =    Ec3(ri,a,index_Pd1,1,XX,...
                    @f_PdPd,X0_PdPd,...
                    @F_Pd,X1_PdPd,...
                    @phi_PdPd,X2_PdPd,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_PdH,rc_1);

    % Pd Central
    Ec_Pd_22 =    Ec3(ri,a,index_Pd2,1,XX,...
                    @f_PdPd,X0_PdPd,...
                    @F_Pd,X1_PdPd,...
                    @phi_PdPd,X2_PdPd,rc_1,...
```

```
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_PdH,rc_1);

    % Pd Central
    Ec_Pd_32 =    Ec3(ri,a,index_Pd3,1,XX,...
                    @f_PdPd,X0_PdPd,...
                    @F_Pd,X1_PdPd,...
                    @phi_PdPd,X2_PdPd,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_PdH,rc_1);

     % Ag Central
    Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
                   @f_PdPd,X0_AgAg,...
                    @F_Pd,X1_AgAg,...
                   @phi_PdPd,X2_AgAg,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_AgH,rc_1);

     %  H Central
    Ec_H_12 =    Ec3(ri,a,index_H1,3,ZZ,...
                    @f_HH,X0_HH,...
                     @F_H,X22_HH,...
                    @phi_HH,X22_HH,rc_1,...
                    1,XX,@f_PdPd,X0_PdPd,...
                     @phi_PdH,X4_PdH,rc_1,...
                     2,YY,@f_PdPd,X0_AgAg,...
                     @phi_PdH,X4_AgH,rc_1);

     %  H Central
    Ec_H_22 =    Ec3(ri,a,index_H2,3,ZZ,...
                    @f_HH,X0_HH,...
                     @F_H,X22_HH,...
                    @phi_HH,X22_HH,rc_1,...
                    1,XX,@f_PdPd,X0_PdPd,...
                     @phi_PdH,X4_PdH,rc_1,...
                     2,YY,@f_PdPd,X0_AgAg,...
                     @phi_PdH,X4_AgH,rc_1);


    Ec_Pd075Ag025H050_O1_O2    = ( xx * Ec_Pd_12 + xx * Ec_Pd_22 + xx *
Ec_Pd_32 + yy * Ec_Ag_12 + zz * Ec_H_12 + zz * Ec_H_22)/(3*xx + yy + 2*zz);


%************************************************************************
% For Pd075Ag025H025_O2_O2
%************************************************************************
    global ri_PdAgH050_O2_O2;
    global index_PdAgH050_O2_O2_Pd1
```

```
global index_PdAgH050_O2_O2_Pd2
global index_PdAgH050_O2_O2_Pd3
global index_PdAgH050_O2_O2_Ag
global index_PdAgH050_O2_O2_H1
global index_PdAgH050_O2_O2_H3

ri = ri_PdAgH050_O2_O2;
index_Pd1 = index_PdAgH050_O2_O2_Pd1;
index_Pd2 = index_PdAgH050_O2_O2_Pd2;
index_Pd3 = index_PdAgH050_O2_O2_Pd3;
index_Ag = index_PdAgH050_O2_O2_Ag;
index_H1  = index_PdAgH050_O2_O2_H1;
index_H3  = index_PdAgH050_O2_O2_H3;

global a_PdAgH050_O2_O2;
a = a_PdAgH050_O2_O2;


XX = 1;
YY = 1;
ZZ = 1;

xx = 0.25;
yy = 0.25;
zz = 0.25;

% Pd Central
Ec_Pd_12 =    Ec3(ri,a,index_Pd1,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);

% Pd Central
Ec_Pd_22 =    Ec3(ri,a,index_Pd2,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
% Pd Central
Ec_Pd_32 =    Ec3(ri,a,index_Pd3,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
```

```
    % Ag Central
    Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
                   @f_PdPd,X0_AgAg,...
                    @F_Pd,X1_AgAg,...
                   @phi_PdPd,X2_AgAg,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_AgH,rc_1);

     % H Central
    Ec_H_12 =    Ec3(ri,a,index_H1,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

     % H Central
    Ec_H_22 =    Ec3(ri,a,index_H3,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);


    Ec_Pd075Ag025H050_O2_O2   = ( xx * Ec_Pd_12 + xx * Ec_Pd_22 + xx *
Ec_Pd_32 + yy * Ec_Ag_12 + zz * Ec_H_12 + zz * Ec_H_12)/(3*xx + yy + 2*zz);

%************************************************************************
% For Pd075Ag025H050_TE1_TE2
%************************************************************************

    global ri_PdAgH050_TE1_TE2;
    global index_PdAgH050_TE1_TE2_Pd1
    global index_PdAgH050_TE1_TE2_Pd2
    global index_PdAgH050_TE1_TE2_Pd3
    global index_PdAgH050_TE1_TE2_Ag
    global index_PdAgH050_TE1_TE2_H_TE1
    global index_PdAgH050_TE1_TE2_H_TE2

    ri = ri_PdAgH050_TE1_TE2;
    index_Pd1 = index_PdAgH050_TE1_TE2_Pd1;
    index_Pd2 = index_PdAgH050_TE1_TE2_Pd2;
    index_Pd3 = index_PdAgH050_TE1_TE2_Pd3;
    index_Ag = index_PdAgH050_TE1_TE2_Ag;
    index_H_TE1  = index_PdAgH050_TE1_TE2_H_TE1;
    index_H_TE2  = index_PdAgH050_TE1_TE2_H_TE2;

    global a_PdAgH050_TE1_TE2
```

```
a = a_PdAgH050_TE1_TE2;

XX = 1;
YY = 1;
ZZ = 1;

xx = 0.25;
yy = 0.25;
zz = 0.25;

% Pd Central
Ec_Pd_12 =    Ec3(ri,a,index_Pd1,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
% Pd Central
Ec_Pd_22 =    Ec3(ri,a,index_Pd2,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
% Pd Central
Ec_Pd_32 =    Ec3(ri,a,index_Pd3,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);

% Ag Central
Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
               @f_PdPd,X0_AgAg,...
                @F_Pd,X1_AgAg,...
               @phi_PdPd,X2_AgAg,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_AgH,rc_1);

 % H Central
Ec_H_12 =    Ec3(ri,a,index_H_TE1,3,ZZ,...
               @f_HH,X0_HH,...
                @F_H,X22_HH,...
               @phi_HH,X22_HH,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdH,X4_PdH,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
```

```
                    @phi_PdH,X4_AgH,rc_1);

     % H Central
    Ec_H_22 =    Ec3(ri,a,index_H_TE2,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

    Ec_Pd075Ag025H050_TE1_TE2   = ( xx*Ec_Pd_12 + xx*Ec_Pd_22 + xx*Ec_Pd_32 +
yy * Ec_Ag_12 + zz * Ec_H_12 + zz * Ec_H_22)/(3*xx + yy + 2*zz);



%************************************************************************
% For Pd075Ag025H050_TE1_TE8
%************************************************************************

    global ri_PdAgH050_TE1_TE8;
    global index_PdAgH050_TE1_TE8_Pd
    global index_PdAgH050_TE1_TE8_Ag
    global index_PdAgH050_TE1_TE8_H_TE1
    global index_PdAgH050_TE1_TE8_H_TE8


    ri = ri_PdAgH050_TE1_TE8;
    index_Pd = index_PdAgH050_TE1_TE8_Pd;
    index_Ag = index_PdAgH050_TE1_TE8_Ag;
    index_H_TE1  = index_PdAgH050_TE1_TE8_H_TE1;
    index_H_TE8  = index_PdAgH050_TE1_TE8_H_TE8;

    global a_PdAgH050_TE1_TE8
    a = a_PdAgH050_TE1_TE8;

    XX = 1;
    YY = 1;
    ZZ = 1;

    xx = 0.75;
    yy = 0.25;
    zz = 0.50;
    zzz = 0.25;

    % Pd Central
    Ec_Pd_12 =    Ec3(ri,a,index_Pd,1,XX,...
                   @f_PdPd,X0_PdPd,...
                   @F_Pd,X1_PdPd,...
                   @phi_PdPd,X2_PdPd,rc_1,...
                   2,YY,@f_PdPd,X0_AgAg,...
                   @phi_PdAg,X5_PdAg,rc_1,...
                   3,ZZ,@f_HH,X0_HH,...
```

```
                    @phi_PdH,X4_PdH,rc_1);
    % Ag Central
    Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
                   @f_PdPd,X0_AgAg,...
                    @F_Pd,X1_AgAg,...
                   @phi_PdPd,X2_AgAg,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_AgH,rc_1);

     % H Central
    Ec_H_12 =    Ec3(ri,a,index_H_TE1,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

     % H Central
    Ec_H_22 =    Ec3(ri,a,index_H_TE8,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

    Ec_Pd075Ag025H050_TE1_TE8   = ( xx * Ec_Pd_12  + yy * Ec_Ag_12 + zzz *
Ec_H_12+ zzz * Ec_H_22)/(xx + yy + zzz + zzz);



%**************************************************************************
% For Pd075Ag025H050_TE3_TE4
%**************************************************************************

    global ri_PdAgH050_TE3_TE4
    global index_PdAgH050_TE3_TE4_Pd
    global index_PdAgH050_TE3_TE4_Ag
    global index_PdAgH050_TE3_TE4_H_TE3
    global index_PdAgH050_TE3_TE4_H_TE4

    ri = ri_PdAgH050_TE3_TE4;
    index_Pd = index_PdAgH050_TE3_TE4_Pd;
    index_Ag = index_PdAgH050_TE3_TE4_Ag;
    index_H_TE3  = index_PdAgH050_TE3_TE4_H_TE3;
    index_H_TE4  = index_PdAgH050_TE3_TE4_H_TE4;

    a = a_PdAgH050_TE1_TE8;
```

```
    XX = 1;
    YY = 1;
    ZZ = 1;

    xx = 0.75;
    yy = 0.25;
    zz = 0.25;
    zzz = 0.25;


    % Pd Central
    Ec_Pd_12 =    Ec3(ri,a,index_Pd,1,XX,...
                    @f_PdPd,X0_PdPd,...
                    @F_Pd,X1_PdPd,...
                    @phi_PdPd,X2_PdPd,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_PdH,rc_1);
    % Ag Central
    Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
                   @f_PdPd,X0_AgAg,...
                    @F_Pd,X1_AgAg,...
                   @phi_PdPd,X2_AgAg,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_AgH,rc_1);

     % H Central
    Ec_H_12 =    Ec3(ri,a,index_H_TE3,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

     % H Central
    Ec_H_22 =    Ec3(ri,a,index_H_TE4,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);


    Ec_Pd075Ag025H050_TE3_TE4   = ( xx * Ec_Pd_12  + yy * Ec_Ag_12 + zz *
Ec_H_12 + zzz * Ec_H_22)/(xx + yy + zz + zzz);


%*********************************************************************
```

```
% For Pd075Ag025H075_O1_O2_O2
%*************************************************************************
    global ri_PdAgH075_O1_O2_O2;
    global index_PdAgH075_O1_O2_O2_Pd1
    global index_PdAgH075_O1_O2_O2_Pd2
    global index_PdAgH075_O1_O2_O2_Pd3
    global index_PdAgH075_O1_O2_O2_Ag
    global index_PdAgH075_O1_O2_O2_H1
    global index_PdAgH075_O1_O2_O2_H2
    global index_PdAgH075_O1_O2_O2_H3

    ri = ri_PdAgH075_O1_O2_O2;
    index_Pd1 = index_PdAgH075_O1_O2_O2_Pd1;
    index_Pd2 = index_PdAgH075_O1_O2_O2_Pd2;
    index_Pd3 = index_PdAgH075_O1_O2_O2_Pd3;
    index_Ag = index_PdAgH075_O1_O2_O2_Ag;
    index_H1  = index_PdAgH075_O1_O2_O2_H1;
    index_H2  = index_PdAgH075_O1_O2_O2_H2;
    index_H3  = index_PdAgH075_O1_O2_O2_H3;

    global a_PdAgH075_O1_O2_O2;
    a = a_PdAgH075_O1_O2_O2;

    XX = 1;
    YY = 1;
    ZZ = 1;

    xx = 0.25;
    yy = 0.25;
    zz = 0.25;


    % Pd Central
    Ec_Pd_12 =    Ec3(ri,a,index_Pd1,1,XX,...
                  @f_PdPd,X0_PdPd,...
                  @F_Pd,X1_PdPd,...
                  @phi_PdPd,X2_PdPd,rc_1,...
                  2,YY,@f_PdPd,X0_AgAg,...
                  @phi_PdAg,X5_PdAg,rc_1,...
                  3,ZZ,@f_HH,X0_HH,...
                  @phi_PdH,X4_PdH,rc_1);
    % Pd Central
    Ec_Pd_22 =    Ec3(ri,a,index_Pd2,1,XX,...
                  @f_PdPd,X0_PdPd,...
                  @F_Pd,X1_PdPd,...
                  @phi_PdPd,X2_PdPd,rc_1,...
                  2,YY,@f_PdPd,X0_AgAg,...
                  @phi_PdAg,X5_PdAg,rc_1,...
                  3,ZZ,@f_HH,X0_HH,...
                  @phi_PdH,X4_PdH,rc_1);
    % Pd Central
    Ec_Pd_32 =    Ec3(ri,a,index_Pd3,1,XX,...
                  @f_PdPd,X0_PdPd,...
                  @F_Pd,X1_PdPd,...
                  @phi_PdPd,X2_PdPd,rc_1,...
                  2,YY,@f_PdPd,X0_AgAg,...
```

```
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_PdH,rc_1);

     % Ag Central
    Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
                   @f_PdPd,X0_AgAg,...
                    @F_Pd,X1_AgAg,...
                   @phi_PdPd,X2_AgAg,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_AgH,rc_1);

     %  H Central
    Ec_H_12 =    Ec3(ri,a,index_H1,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

      %  H Central
    Ec_H_22 =    Ec3(ri,a,index_H2,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

      %  H Central
    Ec_H_32 =    Ec3(ri,a,index_H3,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

    Ec_Pd075Ag025H075_O1_O2_O2   = ( xx*Ec_Pd_12 + xx*Ec_Pd_22 + xx*Ec_Pd_32
+ yy * Ec_Ag_12 + zz * Ec_H_12 + zz * Ec_H_22 + zz * Ec_H_32)/(3*xx + yy +
3*zz);




%************************************************************************
% For Pd075Ag025H075_O2_O2_O2
%************************************************************************
    global ri_PdAgH075_O2_O2_O2;
```

```
global index_PdAgH075_O2_O2_O2_Pd1
global index_PdAgH075_O2_O2_O2_Pd2
global index_PdAgH075_O2_O2_O2_Pd3
global index_PdAgH075_O2_O2_O2_Ag
global index_PdAgH075_O2_O2_O2_H1
global index_PdAgH075_O2_O2_O2_H2
global index_PdAgH075_O2_O2_O2_H3

ri = ri_PdAgH075_O2_O2_O2;
index_Pd = index_PdAgH075_O2_O2_O2_Pd1;
index_Pd = index_PdAgH075_O2_O2_O2_Pd2;
index_Pd = index_PdAgH075_O2_O2_O2_Pd3;
index_Ag = index_PdAgH075_O2_O2_O2_Ag;
index_H  = index_PdAgH075_O2_O2_O2_H1;
index_H  = index_PdAgH075_O2_O2_O2_H2;
index_H  = index_PdAgH075_O2_O2_O2_H3;

global a_PdAgH075_O2_O2_O2;
a = a_PdAgH075_O2_O2_O2;

XX = 1;
YY = 1;
ZZ = 1;

xx = 0.25;
yy = 0.25;
zz = 0.25;

% Pd Central
Ec_Pd_12 =    Ec3(ri,a,index_Pd1,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
% Pd Central
Ec_Pd_22 =    Ec3(ri,a,index_Pd2,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
% Pd Central
Ec_Pd_32 =    Ec3(ri,a,index_Pd3,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
```

```
     % Ag Central
    Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
                   @f_PdPd,X0_AgAg,...
                    @F_Pd,X1_AgAg,...
                   @phi_PdPd,X2_AgAg,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_AgH,rc_1);

     %  H Central
    Ec_H_12 =    Ec3(ri,a,index_H1,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);
     %  H Central
    Ec_H_22 =    Ec3(ri,a,index_H2,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);
     %  H Central
    Ec_H_32 =    Ec3(ri,a,index_H3,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

    Ec_Pd075Ag025H075_O2_O2_O2   = ( xx * Ec_Pd_12 + xx * Ec_Pd_22 + xx *
Ec_Pd_32 + yy * Ec_Ag_12 + zz * Ec_H_12 + zz * Ec_H_22 + zz * Ec_H_32)/(3*xx
+ yy + 3*zz);




%**************************************************************************
% For Pd075Ag025H075_TE1_TE2_TE3
%**************************************************************************

    global ri_PdAgH075_TE1_TE2_TE3
    global index_PdAgH075_TE1_TE2_TE3_Pd1
    global index_PdAgH075_TE1_TE2_TE3_Pd2
    global index_PdAgH075_TE1_TE2_TE3_Pd3
    global index_PdAgH075_TE1_TE2_TE3_Ag
    global index_PdAgH075_TE1_TE2_TE3_H_TE1
```

```
global index_PdAgH075_TE1_TE2_TE3_H_TE2
global index_PdAgH075_TE1_TE2_TE3_H_TE3

ri = ri_PdAgH075_TE1_TE2_TE3;
index_Pd1 = index_PdAgH075_TE1_TE2_TE3_Pd1;
index_Pd2 = index_PdAgH075_TE1_TE2_TE3_Pd2;
index_Pd3 = index_PdAgH075_TE1_TE2_TE3_Pd3;
index_Ag = index_PdAgH075_TE1_TE2_TE3_Ag;
index_H_TE1  = index_PdAgH075_TE1_TE2_TE3_H_TE1;
index_H_TE2  = index_PdAgH075_TE1_TE2_TE3_H_TE2;
index_H_TE3  = index_PdAgH075_TE1_TE2_TE3_H_TE3;

global a_PdAgH075_TE1_TE2_TE3
a = a_PdAgH075_TE1_TE2_TE3;

XX = 1;
YY = 1;
ZZ = 1;

xx = 0.25;
yy = 0.25;
zz = 0.25;


% Pd Central
Ec_Pd_12 =    Ec3(ri,a,index_Pd1,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
% Pd Central
Ec_Pd_22 =    Ec3(ri,a,index_Pd2,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
% Pd Central
Ec_Pd_32 =    Ec3(ri,a,index_Pd3,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);

% Ag Central
Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
               @f_PdPd,X0_AgAg,...
```

```
                    @F_Pd,X1_AgAg,...
                   @phi_PdPd,X2_AgAg,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_AgH,rc_1);

     %  H Central
    Ec_H_12 =    Ec3(ri,a,index_H_TE1,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

     %  H Central
    Ec_H_22 =    Ec3(ri,a,index_H_TE2,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

     %  H Central
    Ec_H_32 =    Ec3(ri,a,index_H_TE3,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

    Ec_Pd075Ag025H075_TE1_TE2_TE3   = ( xx*Ec_Pd_12 + xx*Ec_Pd_22 +
xx*Ec_Pd_32 + yy * Ec_Ag_12 + zz * Ec_H_12 + zz * Ec_H_22 + zz *
Ec_H_32)/(3*xx + yy + 3*zz);



%*************************************************************************
% For Pd075Ag025H075_TE1_TE2_TE8
%*************************************************************************

    global ri_PdAgH075_TE1_TE2_TE8
    global index_PdAgH075_TE1_TE2_TE8_Pd1
    global index_PdAgH075_TE1_TE2_TE8_Pd2
    global index_PdAgH075_TE1_TE2_TE8_Pd3
    global index_PdAgH075_TE1_TE2_TE8_Ag
    global index_PdAgH075_TE1_TE2_TE8_H_TE1
    global index_PdAgH075_TE1_TE2_TE8_H_TE2
    global index_PdAgH075_TE1_TE2_TE8_H_TE8
```

```
ri = ri_PdAgH075_TE1_TE2_TE8;
index_Pd1 = index_PdAgH075_TE1_TE2_TE8_Pd1;
index_Pd2 = index_PdAgH075_TE1_TE2_TE8_Pd2;
index_Pd3 = index_PdAgH075_TE1_TE2_TE8_Pd3;
index_Ag = index_PdAgH075_TE1_TE2_TE8_Ag;
index_H_TE1  = index_PdAgH075_TE1_TE2_TE8_H_TE1;
index_H_TE2  = index_PdAgH075_TE1_TE2_TE8_H_TE2;
index_H_TE8  = index_PdAgH075_TE1_TE2_TE8_H_TE8;

global a_PdAgH075_TE1_TE2_TE8
a = a_PdAgH075_TE1_TE2_TE8;

XX = 1;
YY = 1;
ZZ = 1;

xx = 0.25;
yy = 0.25;
zz = 0.25;

% Pd Central
Ec_Pd_12 =    Ec3(ri,a,index_Pd1,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
% Pd Central
Ec_Pd_22 =    Ec3(ri,a,index_Pd2,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
% Pd Central
Ec_Pd_32 =    Ec3(ri,a,index_Pd3,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);

% Ag Central
Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
               @f_PdPd,X0_AgAg,...
                @F_Pd,X1_AgAg,...
               @phi_PdPd,X2_AgAg,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdAg,X5_PdAg,rc_1,...
```

```
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_AgH,rc_1);

     %  H Central
    Ec_H_12 =    Ec3(ri,a,index_H_TE1,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

     %  H Central
    Ec_H_22 =    Ec3(ri,a,index_H_TE2,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

     %  H Central
    Ec_H_32 =    Ec3(ri,a,index_H_TE8,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

    Ec_Pd075Ag025H075_TE1_TE2_TE8   = ( xx*Ec_Pd_12 + xx*Ec_Pd_22 +
xx*Ec_Pd_32 + yy * Ec_Ag_12 + zz * Ec_H_12 + zz * Ec_H_22 + zz *
Ec_H_32)/(3*xx + yy + 3*zz);


%************************************************************************
% For Pd075Ag025H100_O1_O2_O2_O2
%************************************************************************
    global ri_PdAgH100_O1_O2_O2_O2;
    global index_PdAgH100_O1_O2_O2_O2_Pd1
    global index_PdAgH100_O1_O2_O2_O2_Pd2
    global index_PdAgH100_O1_O2_O2_O2_Pd3
    global index_PdAgH100_O1_O2_O2_O2_Ag
    global index_PdAgH100_O1_O2_O2_O2_H1
    global index_PdAgH100_O1_O2_O2_O2_H2
    global index_PdAgH100_O1_O2_O2_O2_H3
    global index_PdAgH100_O1_O2_O2_O2_H4


    ri = ri_PdAgH100_O1_O2_O2_O2;
    index_Pd1 = index_PdAgH100_O1_O2_O2_O2_Pd1;
    index_Pd2 = index_PdAgH100_O1_O2_O2_O2_Pd2;
```

```
index_Pd3 = index_PdAgH100_O1_O2_O2_O2_Pd3;
index_Ag = index_PdAgH100_O1_O2_O2_O2_Ag;
index_H1  = index_PdAgH100_O1_O2_O2_O2_H1;
index_H2  = index_PdAgH100_O1_O2_O2_O2_H2;
index_H3  = index_PdAgH100_O1_O2_O2_O2_H3;
index_H4  = index_PdAgH100_O1_O2_O2_O2_H4;

global a_PdAgH100_O1_O2_O2_O2;
a = a_PdAgH100_O1_O2_O2_O2;

XX = 1;
YY = 1;
ZZ = 1;

xx = 0.25;
yy = 0.25;
zz = 0.25;

% Pd Central
Ec_Pd_12 =    Ec3(ri,a,index_Pd1,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);

% Pd Central
Ec_Pd_22 =    Ec3(ri,a,index_Pd2,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);

% Pd Central
Ec_Pd_32 =    Ec3(ri,a,index_Pd3,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);

 % Ag Central
Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
               @f_PdPd,X0_AgAg,...
                @F_Pd,X1_AgAg,...
               @phi_PdPd,X2_AgAg,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdAg,X5_PdAg,rc_1,...
```

```
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_AgH,rc_1);

 %  H Central
Ec_H_12 =    Ec3(ri,a,index_H1,3,ZZ,...
               @f_HH,X0_HH,...
                @F_H,X22_HH,...
               @phi_HH,X22_HH,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdH,X4_PdH,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdH,X4_AgH,rc_1);


 %  H Central
Ec_H_22 =    Ec3(ri,a,index_H2,3,ZZ,...
               @f_HH,X0_HH,...
                @F_H,X22_HH,...
               @phi_HH,X22_HH,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdH,X4_PdH,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdH,X4_AgH,rc_1);

 %  H Central
Ec_H_32 =    Ec3(ri,a,index_H3,3,ZZ,...
               @f_HH,X0_HH,...
                @F_H,X22_HH,...
               @phi_HH,X22_HH,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdH,X4_PdH,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdH,X4_AgH,rc_1);

 %  H Central
Ec_H_42 =    Ec3(ri,a,index_H4,3,ZZ,...
               @f_HH,X0_HH,...
                @F_H,X22_HH,...
               @phi_HH,X22_HH,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdH,X4_PdH,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdH,X4_AgH,rc_1);


    Ec_Pd075Ag025H100_O1_O2_O2_O2   = ( xx * Ec_Pd_12 + xx * Ec_Pd_22 + xx *
Ec_Pd_32 + yy * Ec_Ag_12 + zz * Ec_H_12 + zz * Ec_H_22 + zz * Ec_H_32 + zz *
Ec_H_42)/(3*xx + yy + 4*zz);


%**************************************************************************
% For Pd075Ag025H100_TE1_TE2_TE3_TE4
%**************************************************************************

    global ri_PdAgH100_TE1_TE2_TE3_TE4
```

```
global index_PdAgH100_TE1_TE2_TE3_TE4_Pd1
global index_PdAgH100_TE1_TE2_TE3_TE4_Pd2
global index_PdAgH100_TE1_TE2_TE3_TE4_Pd3
global index_PdAgH100_TE1_TE2_TE3_TE4_Ag
global index_PdAgH100_TE1_TE2_TE3_TE4_H_TE1
global index_PdAgH100_TE1_TE2_TE3_TE4_H_TE2
global index_PdAgH100_TE1_TE2_TE3_TE4_H_TE3
global index_PdAgH100_TE1_TE2_TE3_TE4_H_TE4

ri = ri_PdAgH100_TE1_TE2_TE3_TE4;
index_Pd1 = index_PdAgH100_TE1_TE2_TE3_TE4_Pd1;
index_Pd2 = index_PdAgH100_TE1_TE2_TE3_TE4_Pd2;
index_Pd3 = index_PdAgH100_TE1_TE2_TE3_TE4_Pd3;
index_Ag = index_PdAgH100_TE1_TE2_TE3_TE4_Ag;
index_H_TE1  = index_PdAgH100_TE1_TE2_TE3_TE4_H_TE1;
index_H_TE2  = index_PdAgH100_TE1_TE2_TE3_TE4_H_TE2;
index_H_TE3  = index_PdAgH100_TE1_TE2_TE3_TE4_H_TE3;
index_H_TE4  = index_PdAgH100_TE1_TE2_TE3_TE4_H_TE4;

global a_PdAgH100_TE1_TE2_TE3_TE4
a = a_PdAgH100_TE1_TE2_TE3_TE4;

XX = 1;
YY = 1;
ZZ = 1;

xx = 0.25;
yy = 0.25;
zz = 0.25;

% Pd Central
Ec_Pd_12 =    Ec3(ri,a,index_Pd1,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
% Pd Central
Ec_Pd_22 =    Ec3(ri,a,index_Pd2,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
% Pd Central
Ec_Pd_32 =    Ec3(ri,a,index_Pd3,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
```

```
                @phi_PdH,X4_PdH,rc_1);

% Ag Central
Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
               @f_PdPd,X0_AgAg,...
                @F_Pd,X1_AgAg,...
               @phi_PdPd,X2_AgAg,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_AgH,rc_1);

 %  H Central
Ec_H_12 =    Ec3(ri,a,index_H_TE1,3,ZZ,...
               @f_HH,X0_HH,...
                @F_H,X22_HH,...
               @phi_HH,X22_HH,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdH,X4_PdH,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdH,X4_AgH,rc_1);

 %  H Central
Ec_H_22 =    Ec3(ri,a,index_H_TE2,3,ZZ,...
               @f_HH,X0_HH,...
                @F_H,X22_HH,...
               @phi_HH,X22_HH,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdH,X4_PdH,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdH,X4_AgH,rc_1);

 %  H Central
Ec_H_32 =    Ec3(ri,a,index_H_TE3,3,ZZ,...
               @f_HH,X0_HH,...
                @F_H,X22_HH,...
               @phi_HH,X22_HH,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdH,X4_PdH,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdH,X4_AgH,rc_1);

 %  H Central
Ec_H_42 =    Ec3(ri,a,index_H_TE4,3,ZZ,...
               @f_HH,X0_HH,...
                @F_H,X22_HH,...
               @phi_HH,X22_HH,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdH,X4_PdH,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdH,X4_AgH,rc_1);
```

```
    Ec_Pd075Ag025H100_TE1_TE2_TE3_TE4   = ( xx*Ec_Pd_12 + xx*Ec_Pd_22 +
xx*Ec_Pd_32 + yy * Ec_Ag_12 + zz * Ec_H_12 + zz * Ec_H_22 + zz * Ec_H_32 + zz
* Ec_H_42)/(3*xx + yy + 4*zz);


%************************************************************************
% For Pd075Ag025H100_TE1_TE2_TE7_TE8
%************************************************************************

    global ri_PdAgH100_TE1_TE2_TE7_TE8
    global index_PdAgH100_TE1_TE2_TE7_TE8_Pd
    global index_PdAgH100_TE1_TE2_TE7_TE8_Ag
    global index_PdAgH100_TE1_TE2_TE7_TE8_H_TE1
    global index_PdAgH100_TE1_TE2_TE7_TE8_H_TE2
    global index_PdAgH100_TE1_TE2_TE7_TE8_H_TE7
    global index_PdAgH100_TE1_TE2_TE7_TE8_H_TE8

    ri = ri_PdAgH100_TE1_TE2_TE7_TE8;
    index_Pd = index_PdAgH100_TE1_TE2_TE7_TE8_Pd;
    index_Ag = index_PdAgH100_TE1_TE2_TE7_TE8_Ag;
    index_H_TE1  = index_PdAgH100_TE1_TE2_TE7_TE8_H_TE1;
    index_H_TE2  = index_PdAgH100_TE1_TE2_TE7_TE8_H_TE2;
    index_H_TE7  = index_PdAgH100_TE1_TE2_TE7_TE8_H_TE7;
    index_H_TE8  = index_PdAgH100_TE1_TE2_TE7_TE8_H_TE8;

    global a_PdAgH100_TE1_TE2_TE7_TE8
    a = a_PdAgH100_TE1_TE2_TE7_TE8;

    XX = 1;
    YY = 1;
    ZZ = 1;

    xx = 0.75;
    yy = 0.25;
    zz = 0.25;

    % Pd Central
    Ec_Pd_12 =    Ec3(ri,a,index_Pd,1,XX,...
                  @f_PdPd,X0_PdPd,...
                  @F_Pd,X1_PdPd,...
                  @phi_PdPd,X2_PdPd,rc_1,...
                  2,YY,@f_PdPd,X0_AgAg,...
                  @phi_PdAg,X5_PdAg,rc_1,...
                  3,ZZ,@f_HH,X0_HH,...
                  @phi_PdH,X4_PdH,rc_1);
    % Ag Central
    Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
                   @f_PdPd,X0_AgAg,...
                   @F_Pd,X1_AgAg,...
                  @phi_PdPd,X2_AgAg,rc_1,...
                  1,XX,@f_PdPd,X0_PdPd,...
                   @phi_PdAg,X5_PdAg,rc_1,...
                   3,ZZ,@f_HH,X0_HH,...
                   @phi_PdH,X4_AgH,rc_1);
```

```
 %  H Central
Ec_H_12 =    Ec3(ri,a,index_H_TE1,3,ZZ,...
               @f_HH,X0_HH,...
                @F_H,X22_HH,...
               @phi_HH,X22_HH,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdH,X4_PdH,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdH,X4_AgH,rc_1);

 %  H Central
Ec_H_22 =    Ec3(ri,a,index_H_TE2,3,ZZ,...
               @f_HH,X0_HH,...
                @F_H,X22_HH,...
               @phi_HH,X22_HH,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdH,X4_PdH,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdH,X4_AgH,rc_1);

 %  H Central
Ec_H_32 =    Ec3(ri,a,index_H_TE7,3,ZZ,...
               @f_HH,X0_HH,...
                @F_H,X22_HH,...
               @phi_HH,X22_HH,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdH,X4_PdH,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdH,X4_AgH,rc_1);

 %  H Central
Ec_H_42 =    Ec3(ri,a,index_H_TE8,3,ZZ,...
               @f_HH,X0_HH,...
                @F_H,X22_HH,...
               @phi_HH,X22_HH,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdH,X4_PdH,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdH,X4_AgH,rc_1);


Ec_Pd075Ag025H100_TE1_TE2_TE7_TE8   = ( xx * Ec_Pd_12  + yy * Ec_Ag_12 +
zz * Ec_H_12 + zz * Ec_H_22 + zz * Ec_H_32 + zz * Ec_H_42)/(xx + yy + 4*zz);



%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

d1  = (Ec_Pd075Ag025H025_O1 - (Ec_exp_PdAgH025_O1))^2;
d2  = (Ec_Pd075Ag025H025_O2 - (Ec_exp_PdAgH025_O2))^2;
d3  = (Ec_Pd075Ag025H025_TE1 - (Ec_exp_PdAgH025_TE1))^2;
d4  = (Ec_Pd075Ag025H050_O1_O2 - (Ec_exp_PdAgH050_O1_O2))^2;
d5  = (Ec_Pd075Ag025H050_O2_O2 - (Ec_exp_PdAgH050_O2_O2))^2;
d6  = (Ec_Pd075Ag025H050_TE3_TE4 - (Ec_exp_PdAgH050_TE3_TE4))^2;
d7  = (Ec_Pd075Ag025H050_TE1_TE8 - (Ec_exp_PdAgH050_TE1_TE8))^2;
```

```
    d8  = (Ec_Pd075Ag025H075_O1_O2_O2 - (Ec_exp_PdAgH075_O1_O2_O2))^2;
    d9  = (Ec_Pd075Ag025H075_O2_O2_O2 - (Ec_exp_PdAgH075_O2_O2_O2))^2;
    d10  = (Ec_Pd075Ag025H075_TE1_TE2_TE3 - (Ec_exp_PdAgH075_TE1_TE2_TE3))^2;
    d11  = (Ec_Pd075Ag025H075_TE1_TE2_TE8 - (Ec_exp_PdAgH075_TE1_TE2_TE8))^2;
%     d11  = (Ec_Pd075Ag025H075_TE5_TE6_TE7 -
(Ec_exp_PdAgH075_TE5_TE6_TE7))^2;
     d12  = (Ec_Pd075Ag025H100_O1_O2_O2_O2 -
(Ec_exp_PdAgH100_O1_O2_O2_O2))^2;
     d13  = (Ec_Pd075Ag025H100_TE1_TE2_TE3_TE4 -
(Ec_exp_PdAgH100_TE1_TE2_TE3_TE4))^2;
     d14  = (Ec_Pd075Ag025H100_TE1_TE2_TE7_TE8 -
(Ec_exp_PdAgH100_TE1_TE2_TE7_TE8))^2;
%      d14  = (Ec_Pd075Ag025H100_TE1_TE5_TE6_TE7 -
(Ec_exp_PdAgH100_TE1_TE5_TE6_TE7))^2;


    f  =  sqrt( d1 +  d2  +  d3  +  d4  +  d5  +  d6  +  d7 +  d8  +  d9  +
d10  +  d11 + d12 + d13 + d14)

end


%**************************************************************************
%
%**************************************************************************
function [Ec_,dEda_]=
fitProperties_1(r,a,N,index,t1,y,f_11,dfdr_11,d2fdr2_11,X0,...

F,dFdrho,d2Fdrho2,X1,phi_11,dphidr_11,d2phidr2_11,X2,rc_1,...

t2,x,f_22,dfdr_22,d2fdr2_22,X3,phi_12,dphidr_12,d2phidr2_12,X4,rc_2)

    Ec_ =
Ec(r,a,index,t1,y,f_11,X0,F,X1,phi_11,X2,rc_1,t2,x,f_22,X3,phi_12,X4,rc_2);

    dEda_ =
dEda(r,a,index,t1,y,f_11,dfdr_11,X0,dFdrho,X1,dphidr_11,X2,rc_1,...
                        t2,x,f_22,dfdr_22,X3,dphidr_12,X4,rc_2);
end

%**************************************************************************
% Property Functions Used In Parameters Fitting
%**************************************************************************
function [Ec_,dEda_,C11,C12,C44,Bm_,Ev_] = fitProperties(r,a,N,index,...
          t1,y,f_11,dfdr_11,d2fdr2_11,X0,F,dFdrho,d2Fdrho2,X1,phi_11,...
          dphidr_11,d2phidr2_11,X2,rc_1,t2,x,f_22,dfdr_22,d2fdr2_22,X3,...
          phi_12,dphidr_12,d2phidr2_12,X4,rc_2)
    % Cohesive Energy
    Ec_ = Ec(r,a,index,t1,y,f_11,X0,F,X1,phi_11,X2,rc_1,t2,x,f_22,X3,...
             phi_12,X4,rc_2);
    % First Derivative of Ec with respect to a
    dEda_ = dEda(r,a,index,t1,y,f_11,dfdr_11,X0,dFdrho,X1,dphidr_11,X2,...
                 rc_1,t2,x,f_22,dfdr_22,X3,dphidr_12,X4,rc_2);
    % C11 Elastic Constant
    C11 = Cijkl(1,1,1,1,r,a,N,index,t1,y,f_11,dfdr_11,d2fdr2_11,X0,...
```

```
                dFdrho,d2Fdrho2,X1,dphidr_11,d2phidr2_11,X2,rc_1,...

t2,x,f_22,dfdr_22,d2fdr2_22,X3,dphidr_12,d2phidr2_12,X4,rc_2);
    % C12 Elastic Constant
    C12 = Cijkl(1,1,2,2,r,a,N,index,t1,y,f_11,dfdr_11,d2fdr2_11,X0,...
                dFdrho,d2Fdrho2,X1,dphidr_11,d2phidr2_11,X2,rc_1,...

t2,x,f_22,dfdr_22,d2fdr2_22,X3,dphidr_12,d2phidr2_12,X4,rc_2);
    % C44 Elastic Constant
    C44 = Cijkl(2,3,2,3,r,a,N,index,t1,y,f_11,dfdr_11,d2fdr2_11,X0,...
                dFdrho,d2Fdrho2,X1,dphidr_11,d2phidr2_11,X2,rc_1,...

t2,x,f_22,dfdr_22,d2fdr2_22,X3,dphidr_12,d2phidr2_12,X4,rc_2);
    % Bulk Modulus
    Bm_ = Bm(r,a,N,index,t1,y,f_11,dfdr_11,d2fdr2_11,X0,...
             dFdrho,d2Fdrho2,X1,d2phidr2_11,X2,rc_1,t2,x,f_22,dfdr_22,...
             d2fdr2_22,X3,d2phidr2_12,X4,rc_2);
    % Vacancy Formation
    Ev_ = 0;
    % Ev_ = Ev(index,t1,a,phi_11,X2,f_11,X0,F,X1,r,rc_1);
end

%*********************************************************************
% Cohesive Energy for Interstitial Solid Solution
%*********************************************************************
function f = Ec3(ri,a,index,t1,y,f_11,X0,F_11,X1,phi_11,X2,rc_1,t2,x,...
                f_22,X3,phi_12,X4,rc_2,t3,z,f_33,X5,phi_13,X6,rc_3)
    %
    f = 0;
    Rho_12 = 0;
    Phi_12 = 0;
    Rho_13 = 0;
    Phi_13 = 0;
    %
    Rho_11 = y * Rho(index,t1,ri,a,rc_1,f_11,X0);
    Phi_11 = y * Phi(index,t1,ri,a,rc_1,phi_11,X2);
    %
    if nargin >= 19
       Rho_12  = x * Rho(index,t2,ri,a,rc_2,f_22,X3);
       Phi_12  = x * Phi(index,t2,ri,a,rc_2,phi_12,X4);
    end

    if nargin >= 26
       Rho_13  = z * Rho(index,t3,ri,a,rc_3,f_33,X5);
       Phi_13  = z * Phi(index,t3,ri,a,rc_3,phi_13,X6);
    end
    %
    Rho_1 = Rho_11 + Rho_12 + Rho_13;
    F_1 = F_11(Rho_1,X1);
    f = F_1 + 0.5 * Phi_11 + 0.5 * Phi_12 + 0.5 * Phi_13;
end


%*********************************************************************
% Elestic Constants for an Interstitial Solid Solution
%*********************************************************************
```

```
function f = Cijkl(i,j,k,l,r,a,N,index,t1,y,f_11,dfdr_11,d2fdr2_11,X0,...
                   dFdrho,d2Fdrho2,X1,dphidr_11,d2phidr2_11,X2,rc_1,t2,...
                   x,f_22,dfdr_22,d2fdr2_22,X3,dphidr_12,d2phidr2_12,X4,rc_2)
    %
    Rho_12 = 0;
    Vij_12 = 0;
    Vkl_12 = 0;
    Wijkl_12 = 0;
    Bijkl_12 = 0;
    %
    Rho_11   = y * Rho(index,t1,r,a,rc_1,f_11,X0);
    Vij_11   = y * Vij(index,t1,r,a,rc_1,dfdr_11,X0,i,j,0,0);
    Vkl_11   = y * Vij(index,t1,r,a,rc_1,dfdr_11,X0,0,0,k,l);
    Wijkl_11 = y * Wijkl(index,t1,r,a,rc_1,dfdr_11,d2fdr2_11,X0,i,j,k,l);
    Bijkl_11 = y * Bijkl(index,t1,r,a,rc_1,dphidr_11,d2phidr2_11,X2,i,j,k,l);
    %
    if nargin == 31
        Rho_12   = x * Rho(index,t2,r,a,rc_2,f_22,X3);
        Vij_12   = x * Vij(index,t2,r,a,rc_2,dfdr_22,X3,i,j,0,0);
        Vkl_12   = x * Vij(index,t2,r,a,rc_2,dfdr_22,X3,0,0,k,l);
        Wijkl_12 = x * Wijkl(index,t2,r,a,rc_2,dfdr_22,d2fdr2_22,X3,i,j,k,l);
        Bijkl_12 = x *
Bijkl(index,t2,r,a,rc_2,dphidr_12,d2phidr2_12,X4,i,j,k,l);
    end
    Rho_1      = Rho_11 + Rho_12;
    dFdrho_1   = dFdrho(Rho_1,X1);
    d2Fdrho2_1 = d2Fdrho2(Rho_1,X1);
    % Elastic constants Cijkl
    V = (a^3)/N;
    Cijkl = (1/V)*( d2Fdrho2_1*(Vij_11 + Vij_12)*(Vkl_11 + Vkl_12) +...
                    dFdrho_1*(Wijkl_11 + Wijkl_12) + Bijkl_11 + Bijkl_12 );
    f = Cijkl * 1.602176462;
end

%************************************************************************
% Stress for an Interstitial solid solution
%************************************************************************
function f =
S(i,j,r,a,N,index,t1,y,f_11,dfdr_11,X0,dFdrho,X1,dphidr_11,X2,rc_1,...
               t2,x,f_22,dfdr_22,X3,dphidr_12,X4,rc_2)

           Rho_12 = 0;
           Vij_12 = 0;
           Aij_12 = 0;
           %
           Rho_11 = y * Rho(index,t1,r,a,rc_1,f_11,X0);
           Vij_11 = y * Vij(index,t1,r,a,rc_1,dfdr_11,X0,i,j,0,0);
           Aij_11 = y * Aij(index,t1,r,a,rc_1,dphidr_11,X2,i,j,0,0);
           if nargin == 24
               Rho_12 = x * Rho(index,t2,r,a,rc_2,f_22,X3);
               Vij_12 = x * Vij(index,t2,r,a,rc_2,dfdr_22,X3,i,j,0,0);
               Aij_12 = x * Aij(index,t2,r,a,rc_2,dphidr_12,X4,i,j,0,0);
           end
           Rho_1     = Rho_11 + Rho_12;
           dFdrho_1  = dFdrho(Rho_1,X1);
           V = (a^3)/N;
```

```
            f = (1/V) * (Aij_11 + Aij_12 + dFdrho_1 * (Vij_11 + Vij_12));
end

%************************************************************************
% First derivative of Cohesive Energy with respect to a for an
% Interstitial solid solution
%************************************************************************
function  f =
dEda(ri,a,index,t1,y,f_11,dfdr_11,X0,dFdrho,X1,dphidr_11,X2,rc_1,...
                  t2,x,f_22,dfdr_22,X3,dphidr_12,X4,rc_2)
     %
     Rho_12    = 0;
     dRhoda_12 = 0;
     dPhida_12 = 0;
     %
     Rho_11    = y * Rho(index,t1,ri,a,rc_1,f_11,X0);
     dRhoda_11 = y * dRhoda(index,t1,ri,a,rc_1,dfdr_11,X0);
     dPhida_11 = y * dPhida(index,t1,ri,a,rc_1,dphidr_11,X2);
     %
     if nargin == 21
        Rho_12    = x * Rho(index,t2,ri,a,rc_2,f_22,X3);
        dRhoda_12 = x * dRhoda(index,t2,ri,a,rc_2,dfdr_22,X3);
        dPhida_12 = x * dPhida(index,t2,ri,a,rc_2,dphidr_12,X4);
     end
     %
     Rho_1     = Rho_11    + Rho_12;
     dRhoda_1  = dRhoda_11 + dRhoda_12;
     dFda_1 = dFda(dFdrho,Rho_1,dRhoda_1,X1);
     f = dFda_1 + 0.5 * dPhida_11 + 0.5 * dPhida_12;
end

%************************************************************************
% Second derivative of Cohesive Energy with respect to a for Interstitial
% Solid Solution
%************************************************************************
function  f = d2Eda2(ri,a,index,t1,y,f_11,dfdr_11,d2fdr2_11,X0,...
                     dFdrho,d2Fdrho2,X1,d2phidr2_11,X2,rc_1,t2,x,...
                     f_22,dfdr_22,d2fdr2_22,X3,d2phidr2_12,X4,rc_2)
     %
     Rho_12      = 0;
     dRhoda_12   = 0;
     d2Rhoda2_12 = 0;
     d2Phida2_12 = 0;
     %
     Rho_11      = y * Rho(index,t1,ri,a,rc_1,f_11,X0);
     dRhoda_11   = y * dRhoda(index,t1,ri,a,rc_1,dfdr_11,X0);
     d2Rhoda2_11 = y * d2Rhoda2(index,t1,ri,a,rc_1,d2fdr2_11,X0);
     d2Phida2_11 = y * d2Phida2(index,t1,ri,a,rc_1,d2phidr2_11,X2);
     %
     if nargin == 24
        Rho_12      = x * Rho(index,t2,ri,a,rc_2,f_22,X3);
        dRhoda_12   = x * dRhoda(index,t2,ri,a,rc_2,dfdr_22,X3);
        d2Rhoda2_12 = x * d2Rhoda2(index,t2,ri,a,rc_2,d2fdr2_22,X3);
        d2Phida2_12 = x * d2Phida2(index,t2,ri,a,rc_2,d2phidr2_12,X4);
     end
     Rho_1       = Rho_11 + Rho_12;
```

```
     dRhoda_1    = dRhoda_11 + dRhoda_12;
     d2Rhoda2_1  = d2Rhoda2_11 + d2Rhoda2_12;
     d2Fda2_1 = d2Fda2(dFdrho,d2Fdrho2,Rho_1,dRhoda_1,d2Rhoda2_1,X1);
     f = d2Fda2_1 + 0.5 * d2Phida2_11 + 0.5 * d2Phida2_12;
end

%**************************************************************************
% Bulk modulous
%**************************************************************************
function f = Bm(ri,a,N,index,t1,y,f_11,dfdr_11,d2fdr2_11,X0,...
                       dFdrho,d2Fdrho2,X1,d2phidr2_11,X2,rc_1,t2,x,f_22,...
                       dfdr_22,d2fdr2_22,X3,d2phidr2_12,X4,rc_2)
    d2Eda2_ = 0;
    %
    if nargin == 16
       d2Eda2_ = d2Eda2(ri,a,index,t1,y,f_11,dfdr_11,d2fdr2_11,X0,...
                           dFdrho,d2Fdrho2,X1,d2phidr2_11,X2,rc_1);
    end
    if nargin == 25
         d2Eda2_ = d2Eda2(ri,a,index,t1,y,f_11,dfdr_11,d2fdr2_11,X0,...
                       dFdrho,d2Fdrho2,X1,d2phidr2_11,X2,rc_1,t2,x,f_22,...
                       dfdr_22,d2fdr2_22,X3,d2phidr2_12,X4,rc_2);
    end
    V = a^3/N;
    f = ((a)^2/(9.0*V))*d2Eda2_;
    f = f * 1.602176462;
end

%**************************************************************************
% Vacancy formation Energy
%**************************************************************************
function f = Ev(index,type,a,phi,X0,f,X1,F,X2,rij,rc)
    sum0 = 0;
    sum1 = 0;
    sum2 = 0;
    rho = Rho(index,type,rij,a,rc,f,X1);
    n = length(rij(:,1));
    for i = 1:n
        id = rij(i,1);
        if i ~= index && id == type
            l1 = (a/2)*rij(i,2);
            l2 = (a/2)*rij(i,3);
            l3 = (a/2)*rij(i,4);
            ri = sqrt(l1^2+l2^2+l3^2);
            if ri <= rc
                q0 = phi(ri,X0);
                f1 = f(ri,X1);
                sum0 = sum0 + q0;
                sum1 = sum1 + F(rho,X2);
                sum2 = sum2 + F(rho-f1,X2);
            end
        end
    end
    f = -0.5 * sum0 + sum2 - sum1;
end
```

```
%************************************************************************
% The Embedding Atom Model EAM
%************************************************************************

% Total pair potential energy for a central atom as function of interatomic
% distance
% index: array index of central atom
% type : Id of atom type
% ri   : atoms position in Lattice Crystal Structure
% a    : the lattice constant
% rc   : the cuttof radius
% phi  : the pair potential function
% X    : array of pair potential parameters
function f = Phi(index,type,ri,a,rc,phi,X)
    f = Sumfunc(index,type,ri,a,rc,phi,X);
end

% Total first derivative of pair potential with respect to interatomic
% distance rij
function f = dPhidr(index,type,ri,a,rc,dphidr,X)
    f = Sumfunc(index,type,ri,a,rc,dphidr,X);
end

% Total second derivative of pair potential with respect to interatomic
% distance rij
function f = d2Phidr2(index,type,ri,a,rc,d2phidr2,X)
    f = Sumfunc(index,type,ri,a,rc,d2phidr2,X);
end

% Total first derivative of pair potential with respect to a
function f = dPhida(index,type,ri,a,rc,dphidr,X)
    f = SumfuncM(index,type,ri,a,rc,dphidr,X);
end

% Total second derivative of pair potential with respect to a
function f = d2Phida2(index,type,ri,a,rc,d2phidr2,X)
    f = SumfuncM2(index,type,ri,a,rc,d2phidr2,X);
end

% Total Electron Density for a central atom as function of interatomic
% distance
% index: array index of central atom
% type : Id of atom type
% ri   : atoms position in Lattice Crystal Structure
% a    : the lattice constant
% rc   : the cuttof radius
% f    : the atomic density function
% X    : array of atomic density function parameters
function f = Rho(index,type,ri,a,rc,f,X)
    f = Sumfunc(index,type,ri,a,rc,f,X);
end

% Total first derivative of electron density with respect to interatomic
% distance rij
function f = dRhodr(index,type,ri,a,rc,dfdr,X)
```

```
    f = Sumfunc(index,type,ri,a,rc,dfdr,X);
end

% Total second derivative of electron density with respect to interatomic
% distance rij
function f = d2Rhodr2(index,type,ri,a,rc,d2fdr2,X)
    f = Sumfunc(index,type,ri,a,d2fdr2,X);
end

% Total first derivative of electron density with respect a
function f = dRhoda(index,type,ri,a,rc,dfdr,X)
    f = SumfuncM(index,type,ri,a,rc,dfdr,X);
end

% Total second derivative of electron density with respect a
function f = d2Rhoda2(index,type,ri,a,rc,d2fdr2,X)
    f = SumfuncM2(index,type,ri,a,rc,d2fdr2,X);
end

% Total first derivative of electron density with respect ri,rj,rk, and rl
function f = dRhodrij(index,type,ri,a,rc,dfdr,X,i,j,k,l)
     f = Sumfuncrijkl(index,type,ri,a,rc,dfdr,X,i,j,k,l,1);
end

% second derivative of electron density of rij
function f = d2Rhodr2ijkl(index,type,ri,a,rc,dfdr,X,i,j,k,l)
     f = Sumfuncrijkl(index,type,ri,a,rc,dfdr,X,i,j,k,l,2);
end

% First derivative of Atomic Electron Density with respect to ri
function f = dfdri(index,type,ri,a,rc,dfdr,X,i,j,k,l)
     f = Sumfuncrijkl(index,type,ri,a,rc,dfdr,X,i,j,k,l,1);
end

%  First derivative of Atomic Electron Density with respect to ri,rj,rk
%  and rl
function f = dfdrijkl(index,type,ri,a,rc,dfdr,X,i,j,k,l)
     f = Sumfuncrijkl(index,type,ri,a,rc,dfdr,X,i,j,k,l,3);
end

%  Second derivative of Atomic Electron Density with respect to ri,rj,rk
%  and rl
function f = d2fdr2ijkl(index,type,ri,a,rc,d2fdr2,X,i,j,k,l)
     f = Sumfuncrijkl(index,type,ri,a,rc,d2fdr2,X,i,j,k,l,2);
end

% First derivative of pair potential with respect to ri
function f = dphidri(index,type,ri,a,rc,dphidr,X,i,j,k,l)
     f = Sumfuncrijkl(index,type,ri,a,rc,dphidr,X,i,j,k,l,1);
end

% First derivative of pair potential with respect to ri,rj,rk and rl
function f = dPhidrijkl(index,type,ri,a,rc,dphidr,X,i,j,k,l)
     f = Sumfuncrijkl(index,type,ri,a,rc,dphidr,X,i,j,k,l,3);
end
```

```
% Second derivative of pair potential with respect to ri,rj,rk and rl
function f = d2Phidr2ijkl(index,type,ri,a,rc,d2phidr2,X,i,j,k,l)
     f = Sumfuncrijkl(index,type,ri,a,rc,d2phidr2,X,i,j,k,l,2);
end

% Function Used in the calculations of Elastic Constants at equlibrum
function f = Vij(index,type,ri,a,rc,dfdr,X,i,j,k,l)
    f = delta(i,j) * delta(k,l)*dfdri(index,type,ri,a,rc,dfdr,X,i,j,k,l);
end

% Function Used in the calculations of Elastic Constants at equlibrum
function f = Wijkl(index,type,ri,a,rc,dfdr,d2fdr2,X,i,j,k,l)
    f = d2fdr2ijkl(index,type,ri,a,rc,d2fdr2,X,i,j,k,l) - ...
        dfdrijkl(index,type,ri,a,rc,dfdr,X,i,j,k,l)+ ...
        delta(i,l) * dfdri(index,type,ri,a,rc,dfdr,X,0,0,k,l);
end

% Function Used in the calculations of Elastic Constants at equlibrum
function f = Bijkl(index,type,ri,a,rc,dphidr,d2phidr2,X,i,j,k,l)
     f = d2Phidr2ijkl(index,type,ri,a,rc,d2phidr2,X,i,j,k,l) - ...
           dPhidrijkl(index,type,ri,a,rc,dphidr,X,i,j,k,l)+ ...
         delta(i,l) * dphidri(index,type,ri,a,rc,dphidr,X,0,0,k,l);
     f = 0.5 * f;
end

%*************************************************************************
%   Park Hijazi Pd EAM Potential and Derivatives
%*************************************************************************

% Pd Embedding Energy Function as a function of electron density
function f = F_Pd(rho,X)
    f    = 0;
    Fe   = X(1);
    rhoe = X(2);
    n    = X(3);
    if rho > 0
        f = -Fe*(1-n*log(rho/rhoe))*(rho/rhoe)^n;
    end
end

% First derivative of Pd Embedding Energy Function with respect to
% electron density
function f = dFdrho_Pd(rho,X)
    f    = 0;
    Fe   = X(1);
    rhoe = X(2);
    n    = X(3);
    if rho > 0
        F = F_Pd(rho,X);
        f = (Fe*n/rho)*(rho/rhoe)^n+(n/rhoe)*F*(rho/rhoe)^(-1);
    end
end

% Second derivative of Pd Embedding Energy Function with respect to
```

```
% electron density
function f = d2Fdrho2_Pd(rho,X)
    f    = 0;
    Fe   = X(1);
    rhoe = X(2);
    n    = X(3);
    if rho > 0
        F  = F_Pd(rho,X);
        dF = dFdrho_Pd(rho,X);
        f  = -(n/rho^2)*Fe*(rho/rhoe)^n+(n^2/rho)*(1/rhoe)*Fe*(rho/rhoe)^(n-
1)+...
            (n/rhoe)*dF*(rho/rhoe)^(-1)-(n/rhoe^2)*F*(rho/rhoe)^(-2);
    end
end

% First derivative of Pd Embedding Energy Function with respect to rij
function f = dFdr(dFdrho,rho,dRhodr,X)
    f = dFdrho(rho,X) * dRhodr;
end

% Second derivative of Pd Embedding Energy Function with respect to rij
function f = d2Fdr2(dFdrho,d2Fdrho2,rho,dRhodr,d2Rhodr2,X)
    f = d2Fdrho2(rho,X) * dRhodr^2  + dFdrho(rho,X) * d2Rhodr2;
end

% First derivative of Pd Embedding Energy Function with respect to a
function f = dFda(dFdrho,rho,dRhoda,X)
    f = dFdrho(rho,X) * dRhoda;
end

% Second derivative of Pd Embedding Energy Function with respect to a
function f = d2Fda2(dFdrho,d2Fdrho2,rho,dRhoda,d2Rhoda2,X)
    f = d2Fdrho2(rho,X) * dRhoda^2  + dFdrho(rho,X) * d2Rhoda2;
end

% Pd Atomic Electron Density as a function of interatomic distance rij
function f = f_PdPd(rij,X)
    fe = X(1);
    Xi = X(2);
    re = X(3);
    f = fe*exp(-Xi*(rij-re));
end

%  First derivative Pd Atomic Electron Density function with respect to rij
function f = dfdr_PdPd(rij,X)
    fe = X(1);
    Xi = X(2);
    re = X(3);
    f = -fe*Xi*exp(-Xi*(rij-re));
end

%  First derivative Pd Atomic Electron Density function with respect to rij
function f = d2fdr2_PdPd(rij,X)
    fe = X(1);
    Xi = X(2);
```

```
    re = X(3);
    f  = fe*Xi^2*exp(-Xi*(rij-re));
end

% Pd-Pd pair potential as a function of interatomic distance rij
function f = phi_PdPd(rij,X)
    phie = X(1);
    S    = X(2);
    B    = X(3);
    re   = X(4);
    f    = -phie*(1+S*(rij/re-1))*exp(-B*(rij/re-1));
end

% First derivative of Pd-Pd pair potential with respect to rij
function f = dphidr_PdPd(r,X)
    phie = X(1);
    S    = X(2);
    B    = X(3);
    re   = X(4);
    f    = (-1/re)*(phie*S*exp(-B*(r/re-1))+B*phi_PdPd(r,X));
end

% Second derivative of Pd-Pd pair potential with respect to rij
function f = d2phidr2_PdPd(r,X)
    phie = X(1);
    S    = X(2);
    B    = X(3);
    re   = X(4);
    f    = (B/re)*((phie*S/re)*exp(-B*(r/re-1))-dphidr_PdPd(r,X));
end

%*********************************************************************
% Hydrogen EAM Potential and Derivatives
%*********************************************************************

% H-H unnormalized pair potential as a function of interatomic distance rij
function f = phi_HH_u(rij,X)
    DHH     = X(1);
    alphaHH = X(2);
    betaHH  = X(3);
    r0HH    = X(4);
    f = DHH*(betaHH*exp(-alphaHH*(rij-r0HH)) - ...
        alphaHH*exp(-betaHH*(rij-r0HH)));
end

% First derivative of H-H unnormalized pair potential with respect to rij
function f = dphidr_HH_u(r,X)
    DHH     = X(1);
    alphaHH = X(2);
    betaHH  = X(3);
    r0HH    = X(4);
    f = DHH*( -alphaHH * betaHH*exp(-alphaHH*(r-r0HH)) +...
        betaHH*alphaHH*exp(-betaHH*(r-r0HH)) );
end
```

```
% Secon derivative of H-H unnormalized pair potential with respect to rij
function f = d2phidr2_HH_u(r,X)
     DHH     = X(1);
     alphaHH = X(2);
     betaHH  = X(3);
     r0HH    = X(4);
     f = DHH*( (alphaHH^2) * betaHH*exp(-alphaHH*(r-r0HH)) - ...
         (betaHH^2)*alphaHH*exp(-betaHH*(r-r0HH)) );
end

% H-H normalized pair potential as a function of interatomic distance rij
function f = phi_HH(rij,X)
    X2_HH = X(1:1:4);
    X0_HH = X(5:1:6);
    X1_HH = X(7:1:11);
    rho0H = X(11);
    f = phi_HH_u(rij,X2_HH) + 2 * dFdrho_H_u(rho0H,X1_HH) * f_HH(rij,X0_HH);
end

% First derivative of H-H normalized pair potential with respect to rij
function f = dphidr_HH(rij,X)
    X2_HH = X(1:1:4);
    X0_HH = X(5:1:6);
    X1_HH = X(7:1:11);
    rho0H = X(11);
    f = dphidr_HH_u(rij,X2_HH) + 2 * dFdrho_H_u(rho0H,X1_HH) * ...
        dfdr_HH(rij,X0_HH);
end

% Second derivative of H-H normalized pair potential with respect to rij
function f = d2phidr2_HH(rij,X)
    X2_HH = X(1:1:4);
    X0_HH = X(5:1:6);
    X1_HH = X(7:1:11);
    rho0H = X(11);
    f = d2phidr2_HH_u(rij,X2_HH) + 2 * dFdrho_H_u(rho0H,X1_HH) * ...
        d2fdr2_HH(rij,X0_HH);
end

% H Atomic Electron Density as a function of interatomic distance
function f = f_HH(r,X)
    CH = X(1);
    DH = X(2);
    f = CH*exp(-DH*r);
end

% First derivative H Atomic Electron Density with respect to rij
function f = dfdr_HH(r,X)
    CH = X(1);
    DH = X(2);
    f = -DH*CH*exp(-DH*r);
end

% Second derivative H Atomic Electron Density with respect to rij
function f = d2fdr2_HH(r,X)
    CH = X(1);
```

```
    DH = X(2);
    f = (DH^2)*CH*exp(-DH*r);
end

% H unormalized Embedding Energy function
function f = F_H_u(rho,X)
    aH = X(1);
    bH = X(2);
    cH = X(3);
    dH = X(4);
    EH = 0.0540638;
%     EH = 0.055;
    f = -cH * ( (1/(2+dH))*(rho+EH)^(2+dH) - ((aH + bH)/(1+dH))*...
        (rho+EH)^(1+dH) + ((aH * bH)/dH) *(rho+EH)^dH );
end

% First derivative of H unnormalized Embedding function with respect to Rho
function f = dFdrho_H_u(rho,X)
    aH = X(1);
    bH = X(2);
    cH = X(3);
    dH = X(4);
    EH = 0.0540638;
%     EH = 0.055;
    f = -cH * ( (rho+EH)^(1+dH) - (aH + bH) * (rho+EH)^(dH) +...
        (aH * bH) *(rho+EH)^(dH-1) );
end

% Second derivative of H unnormalized Embedding function with respect to Rho
function f = d2Fdrho2_H_u(rho,X)
    aH = X(1);
    bH = X(2);
    cH = X(3);
    dH = X(4);
    EH = 0.0540638;
%     EH = 0.055;
    f = -cH * ( (1+dH)*(rho+EH)^(dH) - dH*(aH + bH) * (rho+EH)^(dH-1) + ...
        (dH-1)*(aH * bH) *(rho+EH)^(dH-2) );
end

% H normalized Embedding Energy function
function f = F_H(rho,X)
    X1_HH = X(7:1:10);
    rho0H = X(11);
    f = F_H_u(rho,X1_HH) - dFdrho_H_u(rho0H,X1_HH) * rho;
end

% First derivative of H normalized Embedding function with respect to Rho
function f = dFdrho_H(rho,X)
    X1_HH = X(7:1:10);
    rho0H = X(11);
    f = dFdrho_H_u(rho,X1_HH) - dFdrho_H_u(rho0H,X1_HH);
end

% Second derivative of H normalized Embedding function with respect to Rho
function f = d2Fdrho2_H(rho,X)
```

```
   X1_HH = X(7:1:10);
   f = d2Fdrho2_H_u(rho,X1_HH);
end

% Pd-H pair potential as a function of interatomic destance rij
function f = phi_PdH(r,X)
        DPdH      = X(1);
        alphaPdH  = X(2);
        betaPdH   = X(3);
        r0PdH     = X(4);
        f = DPdH*( betaPdH*exp(-alphaPdH*(r-r0PdH)) - ...
            alphaPdH*exp(-betaPdH*(r-r0PdH)) );
end

% First derivative of Pd-H pair potential with respect to interatomic
% destance rij
function f = dphidr_PdH(r,X)
        DPdH      = X(1);
        alphaPdH  = X(2);
        betaPdH   = X(3);
        r0PdH     = X(4);
        f = DPdH*( -alphaPdH*betaPdH*exp(-alphaPdH*(r-r0PdH)) +...
            betaPdH*alphaPdH*exp(-betaPdH*(r-r0PdH)) );
end

% Second derivative of Pd-H pair potential with respect to interatomic
% destance rij
function f = d2phidr2_PdH(r,X)
        DPdH      = X(1);
        alphaPdH  = X(2);
        betaPdH   = X(3);
        r0PdH     = X(4);
        f = DPdH*( (alphaPdH^2)*betaPdH*exp(-alphaPdH*(r-r0PdH)) - ...
            (betaPdH^2)*alphaPdH*exp(-betaPdH*(r-r0PdH)) );
end

%*************************************************************************
% Summation function
%*************************************************************************
function f = Sumfunc(index,type,ri,a,rc,func,X)
    f = 0;
    cl1 = (a/2)*ri(index,2);
    cl2 = (a/2)*ri(index,3);
    cl3 = (a/2)*ri(index,4);
    n = length(ri(:,1));
    for i = 1:n
        id = ri(i,1);
        if i ~= index && id == type
            l1 = (a/2)*ri(i,2);
            l2 = (a/2)*ri(i,3);
            l3 = (a/2)*ri(i,4);
            rij = sqrt((l1-cl1)^2 + (l2-cl2)^2 + (l3-cl3)^2);
            if rij <= rc
                f = f + func(rij,X);
            end
        end
```

```
    end
end

function f = SumfuncMP(index,type,ri,a,rc,func,X,p)
    f = 0;
    cl1 = (a/2)*ri(index,2);
    cl2 = (a/2)*ri(index,3);
    cl3 = (a/2)*ri(index,4);
    n = length(ri(:,1));
    for i = 1:n
        id = ri(i,1);
        if i ~= index && id == type
            l1 = (a/2)*ri(i,2);
            l2 = (a/2)*ri(i,3);
            l3 = (a/2)*ri(i,4);
            rij = sqrt((l1-cl1)^2 + (l2-cl2)^2 + (l3-cl3)^2);
            if rij <= rc
                f = f + func(rij,X) * (rij/a)^p;
            end
        end
    end
end

function f = SumfuncM(index,type,ri,a,rc,func,X)
    f = 0;
    cl1 = (a/2)*ri(index,2);
    cl2 = (a/2)*ri(index,3);
    cl3 = (a/2)*ri(index,4);
    n = length(ri(:,1));
    for i = 1:n
        id = ri(i,1);
        if i ~= index && id == type
            l1 = (a/2)*ri(i,2);
            l2 = (a/2)*ri(i,3);
            l3 = (a/2)*ri(i,4);
            rij = sqrt((l1-cl1)^2 + (l2-cl2)^2 + (l3-cl3)^2);
            if rij <= rc
                f = f + func(rij,X) * rij/a;
            end
        end
    end
end

%
function f = SumfuncM2(index,type,ri,a,rc,func,X)
    f = 0;
    cl1 = (a/2)*ri(index,2);
    cl2 = (a/2)*ri(index,3);
    cl3 = (a/2)*ri(index,4);
    n = length(ri(:,1));
    for i = 1:n
        id = ri(i,1);
        if i ~= index && id == type
            l1 = (a/2)*ri(i,2);
            l2 = (a/2)*ri(i,3);
            l3 = (a/2)*ri(i,4);
```

```
            rij = sqrt((l1-cl1)^2 + (l2-cl2)^2 + (l3-cl3)^2);
            if rij <= rc
                  f = f + func(rij,X) * (rij/a)^2;
            end
        end
    end
end

%
function f = Sumfuncrijkl(index,type,r,a,rc,func,X,i,j,k,l,p)
    f = 0;
    ri = 1;
    rj = 1;
    rk = 1;
    rl = 1;
    cl(1) = (a/2)*r(index,2);
    cl(2) = (a/2)*r(index,3);
    cl(3) = (a/2)*r(index,4);
    n = length(r(:,1));
    for ii = 1:n
        id = r(ii,1);
        if ii ~= index && id == type
            l1(1) = (a/2)*r(ii,2);
            l1(2) = (a/2)*r(ii,3);
            l1(3) = (a/2)*r(ii,4);
            rij = sqrt((l1(1)-cl(1))^2 + (l1(2)-cl(2))^2 + (l1(3)-cl(3))^2);
            if rij < rc
                if i ~= 0
                    ri = l1(i)-cl(i);
                end
                if j ~= 0
                    rj = l1(j)-cl(j);
                end
                 if k ~= 0
                    rk = l1(k)-cl(k);
                end
                if l ~= 0
                    rl = l1(l)-cl(l);
                end
                f = f + func(rij,X) * ri*rj*rk*rl/rij^p;
            end
        end
    end
end

%
function f = delta(i,j)
   if(i == j)
        f = 1;
     else
        f = 0;
     end
end


%************************************************************************
```

```
% Expermental data for FCC metals
%************************************************************************
function [afcce,Ecfcce,c11e,c12e,c44e,Bme,Eve,abcce,Ecbcce,cpre] =
parameters(el)
    index = 1;
    elements = ['Ag';'Al';'Au';'Cu';'Ni';'Pd';'Pt';'Hi'];
    for i = 1:length(elements)
        x = elements(i,:);
        if el == x
          index = i;
        end
    end
    afcc =  [4.09;4.05;4.08;3.615;3.52;3.89;3.92;3.38];
    Ecfcc = [2.85;3.36;3.93;3.54;4.45;3.91;5.77;2.119];
    c11 =   [1.24;1.14;1.86;1.70;2.465;2.3412;3.47;0.0];  % from Rayne
    c12 =   [0.934;0.619;1.57;1.225;1.473;1.7614;2.51;0.0];  % from  Rayne
    c44 =   [0.461;0.316;0.42;0.758;1.247;0.7117;0.765;0.05*1.6021766208];  %
    cpr =   [1;1;1;1;1;1;1;0.05*1.6021766208];
    % Unrelaxed
    Evfcc = [1.1;0.866;0.9;1.3;1.70;1.54;1.60;0.0];
    B =     [1.04;0.793;1.67;1.38;1.804;1.9547;2.83;0.5*1.6021766208];
    abcc =  [3.32;3.3;3.18;2.87;2.89;3.16;3.21;0.0];
    Ecbcc = [1;1;1;3.49;1;1;1;1];

    %
    afcce = afcc(index);
    Ecfcce = Ecfcc(index);
    Eve = Evfcc(index);
    c11e = c11(index);
    c12e = c12(index);
    c44e = c44(index);
    Bme = B(index);
    abcce = abcc(index);
    Ecbcce = Ecbcc(index);
    cpre   = cpr(index);
end


%************************************************************************
% Expermental and bb-initio data for PdH
%************************************************************************
function [afcce,Ecfcce,C44fcce,Cprfcce,Bmfcce] = PdH_parameters(el)
    index1 = 1;
    elements = ['Pd000H';'PdH000';'PdH025';'PdH050';'PdH075';...
                'PdH100';'PdHT50';'PdHT75'];
    [n,m]  = size(elements);
    for i = 1:n
        x = elements(i,:);
        if el == x
          index1 = i;
        end
    end
    a  =  [3.38;3.89;3.9421;4.0007;4.0593;4.1179;4.0007;4.0593];
    Ec =  [2.119;3.91;3.64024;3.48770;3.37031;3.26966;3.50000;3.50000];
    C44 = [0.05,0.7117/1.6021766208,0.459,0.42,0.382,0.3440,...
           0.3810,0.3810] * 1.6021766208;  % Pd from Rayne
```

```
    Cpr = [0.05,.2899/1.6021766208,0.055,0.135,0.164,0.146,...
           0.1630,0.1630] * 1.6021766208;  % Pd from Rayne
    Bm =  [0.50,1.9547/1.6021766208,1.05,1.05,1.05,1.05,...
           1.05,1.05] * 1.6021766208;  % Pd from Rayne, Zhou values added
    afcce  = a(index1);
    Ecfcce = Ec(index1);
    C44fcce  = C44(index1);
    Cprfcce  = Cpr(index1);
    Bmfcce  = Bm(index1);
end

 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 %   Pd Cutoff Transformed Functions
 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


function f = phi_PdPd_cut(rij,X)
    global rc_1;
    global NP;
    f = phi_PdPd(rij,X) - phi_PdPd(rc_1,X) + (rc_1/NP) * (1-(rij/rc_1)^NP )*
dphidr_PdPd(rc_1,X);
 end

function f = dphidr_PdPd_cut(rij,X)
    global rc_1;
    global NP;
    f = dphidr_PdPd(rij,X) - (rij/rc_1)^(NP-1) * dphidr_PdPd(rc_1,X);
end

function f = d2phidr2_PdPd_cut(rij,X)
    global rc_1;
    global NP;
    f = d2phidr2_PdPd(rij,X)- ((NP-1)/rc_1) * (rij/rc_1)^(NP-2) *
dphidr_PdPd(rc_1,X);
end

function f = f_PdPd_cut(rij,X)
    global rc_1;
    global NP;
    f = f_PdPd(rij,X) - f_PdPd(rc_1,X) + (rc_1/NP) * (1-(rij/rc_1)^NP) *
dfdr_PdPd(rc_1,X);
end

function f = dfdr_PdPd_cut(rij,X)
    global rc_1;
    global NP;
    f = dfdr_PdPd(rij,X) - (rij/rc_1)^(NP-1) * dfdr_PdPd(rc_1,X);
end

function f = d2fdr2_PdPd_cut(rij,X)
    global rc_1;
    global NP;
    f = d2fdr2_PdPd(rij,X)- ((NP-1)/rc_1) * (rij/rc_1)^(NP-2) *
dfdr_PdPd(rc_1,X);
end
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%   Hydrogen Cutoff Transformed functions
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


function f = phi_HH_cut(rij,X)
    global rc_2;
    global NP2;
    f = phi_HH(rij,X) - phi_HH(rc_2,X) + (rc_2/NP2) * (1-(rij/rc_2)^NP2 )*
dphidr_HH(rc_2,X);
 end

function f = dphidr_HH_cut(rij,X)
    global rc_2;
    global NP2;
    f = dphidr_HH(rij,X) - (rij/rc_2)^(NP2-1) * dphidr_HH(rc_2,X);
end

function f = d2phidr2_HH_cut(rij,X)
    global rc_2;
    global NP2;
    f = d2phidr2_HH(rij,X)- ((NP2-1)/rc_2) * (rij/rc_2)^(NP2-2) *
dphidr_HH(rc_2,X);
end

function f = f_HH_cut(rij,X)
    global rc_2;
    global NP2;
    f = f_HH(rij,X) - f_HH(rc_2,X) + (rc_2/NP2) * (1-(rij/rc_2)^NP2) *
dfdr_HH(rc_2,X);
end

function f = dfdr_HH_cut(rij,X)
    global rc_2;
    global NP2;
    f = dfdr_HH(rij,X) - (rij/rc_2)^(NP2-1) * dfdr_HH(rc_2,X);
end

function f = d2fdr2_HH_cut(rij,X)
    global rc_2;
    global NP2;
    f = d2fdr2_HH(rij,X)- ((NP2-1)/rc_2) * (rij/rc_2)^(NP2-2) *
dfdr_HH(rc_2,X);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%   PdH Cutoff Transformed Pair Potential
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


function f = phi_PdH_cut(rij,X)
    global rc_3;
```

```
    global NP3;
    f = phi_PdH(rij,X) - phi_PdH(rc_3,X) + (rc_3/NP3) * (1-(rij/rc_3)^NP3 )*
dphidr_PdH(rc_3,X);
 end

function f = dphidr_PdH_cut(rij,X)
    global rc_3;
    global NP3;
    f = dphidr_PdH(rij,X) - (rij/rc_3)^(NP3-1) * dphidr_PdH(rc_3,X);
end

function f = d2phidr2_PdH_cut(rij,X)
    global rc_3;
    global NP3;
    f = d2phidr2_PdH(rij,X)- ((NP3-1)/rc_3) * (rij/rc_3)^(NP3-2) *
dphidr_PdH(rc_3,X);
end

 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 %   PdAg Pair Potential
 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

 function f = phi_PdAg(rij,X)
     X0_a = X(1:3);
     X0_b = X(4:6);
     X2_a = X(7:10);
     X2_b = X(11:14);
     f_a = f_PdPd(rij,X0_a);
     f_b = f_PdPd(rij,X0_b);
     phi_a = phi_PdPd(rij,X2_a);
     phi_b = phi_PdPd(rij,X2_b);
     f = .5*( f_b*f_a^-1*phi_a + f_a*f_b^-1*phi_b );
 end

 function f = dphidr_PdAg(rij,X)
     X0_a = X(1:3);
     X0_b = X(4:6);
     X2_a = X(7:10);
     X2_b = X(11:14);
     f_a = f_PdPd(rij,X0_a);
     f_b = f_PdPd(rij,X0_b);
     phi_a = phi_PdPd(rij,X2_a);
     phi_b = phi_PdPd(rij,X2_b);
     dfdr_a = dfdr_PdPd(rij,X0_a);
     dfdr_b = dfdr_PdPd(rij,X0_b);
     dphidr_a = dphidr_PdPd(rij,X2_a);
     dphidr_b = dphidr_PdPd(rij,X2_b);
     f = .5*( ((-1*f_b*dfdr_a*f_a^-2 + dfdr_b*f_a^-1)*phi_a + (f_b*f_a^-
1)*dphidr_a)...
            + ((-1*f_a*dfdr_b*f_b^-2 + dfdr_a*f_b^-1)*phi_b + (f_a*f_b^-
1)*dphidr_b) );
 end

 function f = d2phidr2_PdAg(rij,X)
     X0_a = X(1:3);
     X0_b = X(4:6);
```

```
      X2_a = X(7:10);
      X2_b = X(11:14);
      f_a = f_PdPd(rij,X0_a);
      f_b = f_PdPd(rij,X0_b);
      phi_a = phi_PdPd(rij,X2_a);
      phi_b = phi_PdPd(rij,X2_b);
      dfdr_a = dfdr_PdPd(rij,X0_a);
      dfdr_b = dfdr_PdPd(rij,X0_b);
      dphidr_a = dphidr_PdPd(rij,X2_a);
      dphidr_b = dphidr_PdPd(rij,X2_b);
      d2fdr2_a = d2fdr2_PdPd(rij,X0_a);
      d2fdr2_b = d2fdr2_PdPd(rij,X0_b);
      d2phidr2_a = d2phidr2_PdPd(rij,X2_a);
      d2phidr2_b = d2phidr2_PdPd(rij,X2_b);
      Phi_ab =
phi_ab(f_a,dfdr_a,d2fdr2_a,f_b,dfdr_b,d2fdr2_b,phi_a,dphidr_a,d2phidr2_a,phi_
b,dphidr_b,d2phidr2_b);
      Phi_ba =
phi_ab(f_b,dfdr_b,d2fdr2_b,f_a,dfdr_a,d2fdr2_a,phi_b,dphidr_b,d2phidr2_b,phi_
a,dphidr_a,d2phidr2_a);
      f = .5*(Phi_ab + Phi_ba);
 end

 function f =
phi_ab(f_a,dfdr_a,d2fdr2_a,f_b,dfdr_b,d2fdr2_b,phi_a,dphidr_a,d2phidr2_a,phi_
b,dphidr_b,d2phidr2_b)
    aa = -1*(dfdr_b*dfdr_a*f_a^-2 + f_b*d2fdr2_a*f_a^-2 -
2*f_b*(dfdr_a^2)*f_a^-3)...
     + (d2fdr2_b*f_a^-1 - dfdr_b*f_a^-2*dfdr_a);
    a = aa*phi_a + (-1*f_b*dfdr_a*f_a^-2 + dfdr_b*f_a^-1)*dphidr_a;
    b = dfdr_b*f_a^-1*dphidr_a - f_b*f_a^-2*dfdr_a*dphidr_a + f_b*f_a^-
1*d2phidr2_a;
    f = a + b;
 end
```

Check_Fit.m

```
function Check_Fit(x,plots1,plots2)

    global Ec_exp_PdAgH025_O1;
    global Ec_exp_PdAgH025_O2;
    global Ec_exp_PdAgH025_T;
    global Ec_exp_PdAgH050_O1_O2;
    global Ec_exp_PdAgH050_O2_O2;
    global Ec_exp_PdAgH050_T;
    global Ec_exp_PdAgH075_O1_O2_O2;
    global Ec_exp_PdAgH075_O2_O2_O2;
    global Ec_exp_PdAgH075_T;
    global Ec_exp_PdAgH100_O1_O2_O2_O2;
    global Ec_exp_PdAgH100_T;



    global x_Pd
    global x_Ag
```

```
global S_Ag
global S_Pd

global x_PdH

global rc_1
global rc_2
global rc_3



% Pd Experimental data
[aFcce_Pd,EcFcce_Pd,C11e_Pd,C12e_Pd,C44e_Pd,Bme_Pd,Eve_Pd,aBcce_Pd,...
    EcBcce_Pd] = parameters('Pd');

re = aFcce_Pd/sqrt(2);
Fe = EcFcce_Pd-Eve_Pd;
N  = 4;
V  = aFcce_Pd^3/N;
fe = EcFcce_Pd/V;

fe_S = S_Pd*EcFcce_Pd/V;

% Pd Fitting Parameters
Xi   = x_Pd(1);
phie = x_Pd(2);
S    = x_Pd(3);
B    = x_Pd(4);
n    = x_Pd(5);
rhoe = x_Pd(6);

%  Rho PdPd fitting parameters
X0_PdPd = [fe,Xi,re];
X0_PdPd_S = [fe_S,Xi,re];
%  F PdPd fitting parameters
X1_PdPd = [Fe,rhoe,n];
%  Phi PdPd fitting parametters
X2_PdPd = [phie,S,B,re];


% Ag Experimental data
[aFcce_Ag,EcFcce_Ag,C11e_Ag,C12e_Ag,C44e_Ag,Bme_Ag,Eve_Ag,aBcce_Ag,...
    EcBcce_Ag] = parameters('Ag');

re = aFcce_Ag/sqrt(2);
Fe = EcFcce_Ag-Eve_Ag;
N  = 4;
V  = aFcce_Ag^3/N;
fe = EcFcce_Ag/V;

fe_S = S_Ag*EcFcce_Ag/V;
```

```
% Ag Fitting Parameters
Xi   = x_Ag(1);
phie = x_Ag(2);
S    = x_Ag(3);
B    = x_Ag(4);
n    = x_Ag(5);
rhoe = x_Ag(6);

%  Rho PdPd fitting parameters
X0_AgAg = [fe,Xi,re];
X0_AgAg_S = [fe_S,Xi,re];
%  F PdPd fitting parameters
X1_AgAg = [Fe,rhoe,n];
%  Phi PdPd fitting parametters
X2_AgAg = [phie,S,B,re];

X5_PdAg = [X0_PdPd_S X0_AgAg_S X2_PdPd X2_AgAg];


% Phi_HH
DHH =  x_PdH(1);
aHH =  x_PdH(2);
bHH =  x_PdH(3);
% f_HH
CH = x_PdH(4);
DH = x_PdH(5);
% Phi_PdH
DPdH = x_PdH(6);
aPdH = x_PdH(7);
bPdH = x_PdH(8);
% F_H
aH = x_PdH(9);
bH = x_PdH(10);
cH = x_PdH(11);
dH = x_PdH(12);
%
r0PdH = x_PdH(13);
r0HH =  x_PdH(14);

% f_HH
X0_HH  = [CH,DH];

%     % rho0H calculation
%     ri    = ri_PdH100_OC;
%     index = index_PdH100_OC_1;
%     aFcce_H = 3.38;
%     rho0H = Rho(index,1,ri,aFcce_H,rc_2,@f_HH,X0_HH)
rho0H = 4.903820;

% F HH fitting parameters
X1_HH  = [aH,bH,cH,dH,rho0H];
% Phi HH fitting parametters
X2_HH  = [DHH,aHH,bHH,r0HH];
%
X22_HH = [X2_HH,X0_HH,X1_HH];
```

```
    % Phi PdH fitting parametters
    X4_PdH = [DPdH,aPdH,bPdH,r0PdH];


    % Phi_AgH
    DAgH = x(1);
    aAgH = x(2);
    bAgH = x(3);
    r0AgH = x(4);

    % Phi AgH fitting parametters
    X4_AgH = [DAgH,aAgH,bAgH,r0AgH];


%************************************************************************
% For Pd075Ag025H000
%************************************************************************
    global ri_PdAgH000;
    global index_PdAgH000_Pd
    global index_PdAgH000_Ag

    index_Pd = index_PdAgH000_Pd;
    index_Ag = index_PdAgH000_Ag;
    a = 3.94;

    ri    = ri_PdAgH000;

    XX = 1;
    YY = 1;
    ZZ = 1;

    xx = 0.75;
    yy = 0.25;


    % Pd Central
    Ec_Pd_12 =    Ec3(ri,a,index_Pd,1,XX,...
                  @f_PdPd,X0_PdPd,...
                  @F_Pd,X1_PdPd,...
                  @phi_PdPd,X2_PdPd,rc_1,...
                  2,YY,@f_PdPd,X0_AgAg,...
                  @phi_PdAg,X5_PdAg,rc_1,...
                  3,ZZ,@f_HH,X0_HH,...
                  @phi_PdH,X4_PdH,rc_1);
    % Ag Central
    Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
                  @f_PdPd,X0_AgAg,...
                  @F_Pd,X1_AgAg,...
                  @phi_PdPd,X2_AgAg,rc_1,...
                  1,XX,@f_PdPd,X0_PdPd,...
                  @phi_PdAg,X5_PdAg,rc_1,...
                  3,ZZ,@f_HH,X0_HH,...
                  @phi_PdH,X4_AgH,rc_1);
```

```
Ec_Pd075Ag025H000   = ( xx * Ec_Pd_12  + yy * Ec_Ag_12 )/(xx + yy)

%************************************************************************
% For Pd075Ag025H025_O1
%************************************************************************
    global ri_PdAgH025_O1;
    global index_PdAgH025_O1_Pd1
    global index_PdAgH025_O1_Pd2
    global index_PdAgH025_O1_Pd3
    global index_PdAgH025_O1_Ag
    global index_PdAgH025_O1_H

    ri = ri_PdAgH025_O1;
    index_Pd1 = index_PdAgH025_O1_Pd1;
    index_Pd2 = index_PdAgH025_O1_Pd2;
    index_Pd3 = index_PdAgH025_O1_Pd3;
    index_Ag = index_PdAgH025_O1_Ag;
    index_H  = index_PdAgH025_O1_H;

    global a_PdAgH025_o1;
    a = a_PdAgH025_o1;

    XX = 1;
    YY = 1;
    ZZ = 1;

    xx = 0.25;
    yy = 0.25;
    zz = 0.25;

    % Pd Central
    Ec_Pd_12 =    Ec3(ri,a,index_Pd1,1,XX,...
                    @f_PdPd,X0_PdPd,...
                    @F_Pd,X1_PdPd,...
                    @phi_PdPd,X2_PdPd,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_PdH,rc_1);
    % Pd Central
    Ec_Pd_22 =    Ec3(ri,a,index_Pd2,1,XX,...
                    @f_PdPd,X0_PdPd,...
                    @F_Pd,X1_PdPd,...
                    @phi_PdPd,X2_PdPd,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_PdH,rc_1);
    % Pd Central
    Ec_Pd_32 =    Ec3(ri,a,index_Pd3,1,XX,...
                    @f_PdPd,X0_PdPd,...
                    @F_Pd,X1_PdPd,...
                    @phi_PdPd,X2_PdPd,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
```

```
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_PdH,rc_1);

     % Ag Central
    Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
                   @f_PdPd,X0_AgAg,...
                    @F_Pd,X1_AgAg,...
                   @phi_PdPd,X2_AgAg,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_AgH,rc_1);

     % H Central
    Ec_H_12 =    Ec3(ri,a,index_H,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

    Ec_Pd075Ag025H025_O1   = ( xx * Ec_Pd_12 + xx * Ec_Pd_22 + xx * Ec_Pd_32
+ yy * Ec_Ag_12 + zz * Ec_H_12)/(3*xx + yy + zz)

%************************************************************************
% For Pd075Ag025H025_O2
%************************************************************************
    global ri_PdAgH025_o2;
    global index_PdAgH025_o2_Pd1
    global index_PdAgH025_o2_Pd2
    global index_PdAgH025_o2_Pd3
    global index_PdAgH025_o2_Ag
    global index_PdAgH025_o2_H

    ri = ri_PdAgH025_o2;
    index_Pd1 = index_PdAgH025_o2_Pd1;
    index_Pd2 = index_PdAgH025_o2_Pd2;
    index_Pd3 = index_PdAgH025_o2_Pd3;
    index_Ag = index_PdAgH025_o2_Ag;
    index_H  = index_PdAgH025_o2_H;

    global a_PdAgH025_o2;
    a = a_PdAgH025_o2;


    XX = 1;
    YY = 1;
    ZZ = 1;

    xx = 0.25;
    yy = 0.25;
    zz = 0.25;
```

```
    % Pd Central
    Ec_Pd_12 =    Ec3(ri,a,index_Pd1,1,XX,...
                    @f_PdPd,X0_PdPd,...
                    @F_Pd,X1_PdPd,...
                    @phi_PdPd,X2_PdPd,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_PdH,rc_1);

    % Pd Central
    Ec_Pd_22 =    Ec3(ri,a,index_Pd2,1,XX,...
                    @f_PdPd,X0_PdPd,...
                    @F_Pd,X1_PdPd,...
                    @phi_PdPd,X2_PdPd,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_PdH,rc_1);
    % Pd Central
    Ec_Pd_32 =    Ec3(ri,a,index_Pd3,1,XX,...
                    @f_PdPd,X0_PdPd,...
                    @F_Pd,X1_PdPd,...
                    @phi_PdPd,X2_PdPd,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_PdH,rc_1);

     % Ag Central
    Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
                   @f_PdPd,X0_AgAg,...
                    @F_Pd,X1_AgAg,...
                   @phi_PdPd,X2_AgAg,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_AgH,rc_1);

     % H Central
    Ec_H_12 =    Ec3(ri,a,index_H,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

    Ec_Pd075Ag025H025_O2   = ( xx * Ec_Pd_12 + xx * Ec_Pd_22 + xx * Ec_Pd_32
+ yy * Ec_Ag_12 + zz * Ec_H_12)/(3*xx + yy + zz)

%************************************************************************
% For Pd075Ag025H025_TE1
%************************************************************************
```

```
    global ri_PdAgH025_TE1;
    global index_PdAgH025_TE1_Pd
    global index_PdAgH025_TE1_Ag
    global index_PdAgH025_TE1_H

    ri = ri_PdAgH025_TE1;
    index_Pd = index_PdAgH025_TE1_Pd;
    index_Ag = index_PdAgH025_TE1_Ag;
    index_H  = index_PdAgH025_TE1_H;

    global a_PdAgH025_TE1
    a = a_PdAgH025_TE1;

    XX = 1;
    YY = 1;
    ZZ = 1;

    xx = 0.75;
    yy = 0.25;
    zz = 0.25;

    % Pd Central
    Ec_Pd_12 =    Ec3(ri,a,index_Pd,1,XX,...
                    @f_PdPd,X0_PdPd,...
                    @F_Pd,X1_PdPd,...
                    @phi_PdPd,X2_PdPd,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_PdH,rc_1);
    % Ag Central
    Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
                   @f_PdPd,X0_AgAg,...
                    @F_Pd,X1_AgAg,...
                   @phi_PdPd,X2_AgAg,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_AgH,rc_1);

     % H Central
    Ec_H_12 =    Ec3(ri,a,index_H,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

    Ec_Pd075Ag025H025_TE1   = ( xx * Ec_Pd_12  + yy * Ec_Ag_12 + zz *
Ec_H_12)/(xx + yy + zz)


%*********************************************************************
```

```
% For Pd075Ag025H025_TE8
%*************************************************************************
    global ri_PdAgH025_TE8;
    global index_PdAgH025_TE8_Pd
    global index_PdAgH025_TE8_Ag
    global index_PdAgH025_TE8_H

    ri = ri_PdAgH025_TE8;
    index_Pd = index_PdAgH025_TE8_Pd;
    index_Ag = index_PdAgH025_TE8_Ag;
    index_H  = index_PdAgH025_TE8_H;

    global a_PdAgH025_TE8
    a = a_PdAgH025_TE8;

    XX = 1;
    YY = 1;
    ZZ = 1;

    xx = 0.75;
    yy = 0.25;
    zz = 0.25;

    % Pd Central
    Ec_Pd_12 =    Ec3(ri,a,index_Pd,1,XX,...
                  @f_PdPd,X0_PdPd,...
                  @F_Pd,X1_PdPd,...
                  @phi_PdPd,X2_PdPd,rc_1,...
                  2,YY,@f_PdPd,X0_AgAg,...
                  @phi_PdAg,X5_PdAg,rc_1,...
                  3,ZZ,@f_HH,X0_HH,...
                  @phi_PdH,X4_PdH,rc_1);
    % Ag Central
    Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
                  @f_PdPd,X0_AgAg,...
                  @F_Pd,X1_AgAg,...
                  @phi_PdPd,X2_AgAg,rc_1,...
                  1,XX,@f_PdPd,X0_PdPd,...
                  @phi_PdAg,X5_PdAg,rc_1,...
                  3,ZZ,@f_HH,X0_HH,...
                  @phi_PdH,X4_AgH,rc_1);

     % H Central
    Ec_H_12 =    Ec3(ri,a,index_H,3,ZZ,...
                  @f_HH,X0_HH,...
                  @F_H,X22_HH,...
                  @phi_HH,X22_HH,rc_1,...
                  1,XX,@f_PdPd,X0_PdPd,...
                  @phi_PdH,X4_PdH,rc_1,...
                  2,YY,@f_PdPd,X0_AgAg,...
                  @phi_PdH,X4_AgH,rc_1);

    Ec_Pd075Ag025H025_TE8   = ( xx * Ec_Pd_12  + yy * Ec_Ag_12 + zz *
Ec_H_12)/(xx + yy + zz)
```

```
%*************************************************************************
% For Pd075Ag025H025_TE6
%*************************************************************************

    global ri_PdAgH025_TE6;
    global index_PdAgH025_TE6_Pd
    global index_PdAgH025_TE6_Ag
    global index_PdAgH025_TE6_H

    ri = ri_PdAgH025_TE6;
    index_Pd = index_PdAgH025_TE6_Pd;
    index_Ag = index_PdAgH025_TE6_Ag;
    index_H  = index_PdAgH025_TE6_H;

    global a_PdAgH025_TE6
    a = a_PdAgH025_TE6;

    XX = 1;
    YY = 1;
    ZZ = 1;

    xx = 0.75;
    yy = 0.25;
    zz = 0.25;

    % Pd Central
    Ec_Pd_12 =    Ec3(ri,a,index_Pd,1,XX,...
                    @f_PdPd,X0_PdPd,...
                    @F_Pd,X1_PdPd,...
                    @phi_PdPd,X2_PdPd,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_PdH,rc_1);
    % Ag Central
    Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
                   @f_PdPd,X0_AgAg,...
                    @F_Pd,X1_AgAg,...
                   @phi_PdPd,X2_AgAg,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_AgH,rc_1);

     % H Central
    Ec_H_12 =    Ec3(ri,a,index_H,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);
```

```
    Ec_Pd075Ag025H025_TE6   = ( xx * Ec_Pd_12  + yy * Ec_Ag_12 + zz *
Ec_H_12)/(xx + yy + zz)



%************************************************************************
% For Pd075Ag025H050_O1_O2
%************************************************************************
    global ri_PdAgH050_O1_O2;
    global index_PdAgH050_O1_O2_Pd1
    global index_PdAgH050_O1_O2_Pd2
    global index_PdAgH050_O1_O2_Pd3
    global index_PdAgH050_O1_O2_Ag
    global index_PdAgH050_O1_O2_H_O1
    global index_PdAgH050_O1_O2_H_O2

    ri = ri_PdAgH050_O1_O2;
    index_Pd1 = index_PdAgH050_O1_O2_Pd1;
    index_Pd2 = index_PdAgH050_O1_O2_Pd2;
    index_Pd3 = index_PdAgH050_O1_O2_Pd3;
    index_Ag = index_PdAgH050_O1_O2_Ag;
    index_H1  = index_PdAgH050_O1_O2_H_O1;
    index_H2  = index_PdAgH050_O1_O2_H_O2;

    global a_PdAgH050_O1_O2
    a = a_PdAgH050_O1_O2;



    XX = 1;
    YY = 1;
    ZZ = 1;

    xx = 0.25;
    yy = 0.25;
    zz = 0.25;

    % Pd Central
    Ec_Pd_12 =    Ec3(ri,a,index_Pd1,1,XX,...
                  @f_PdPd,X0_PdPd,...
                  @F_Pd,X1_PdPd,...
                  @phi_PdPd,X2_PdPd,rc_1,...
                  2,YY,@f_PdPd,X0_AgAg,...
                  @phi_PdAg,X5_PdAg,rc_1,...
                  3,ZZ,@f_HH,X0_HH,...
                  @phi_PdH,X4_PdH,rc_1);

    % Pd Central
    Ec_Pd_22 =    Ec3(ri,a,index_Pd2,1,XX,...
                  @f_PdPd,X0_PdPd,...
                  @F_Pd,X1_PdPd,...
                  @phi_PdPd,X2_PdPd,rc_1,...
                  2,YY,@f_PdPd,X0_AgAg,...
                  @phi_PdAg,X5_PdAg,rc_1,...
                  3,ZZ,@f_HH,X0_HH,...
```

```
                @phi_PdH,X4_PdH,rc_1);

    % Pd Central
    Ec_Pd_32 =    Ec3(ri,a,index_Pd3,1,XX,...
                  @f_PdPd,X0_PdPd,...
                  @F_Pd,X1_PdPd,...
                  @phi_PdPd,X2_PdPd,rc_1,...
                  2,YY,@f_PdPd,X0_AgAg,...
                  @phi_PdAg,X5_PdAg,rc_1,...
                  3,ZZ,@f_HH,X0_HH,...
                  @phi_PdH,X4_PdH,rc_1);

     % Ag Central
    Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
                  @f_PdPd,X0_AgAg,...
                   @F_Pd,X1_AgAg,...
                  @phi_PdPd,X2_AgAg,rc_1,...
                  1,XX,@f_PdPd,X0_PdPd,...
                   @phi_PdAg,X5_PdAg,rc_1,...
                   3,ZZ,@f_HH,X0_HH,...
                   @phi_PdH,X4_AgH,rc_1);

     %  H Central
    Ec_H_12 =    Ec3(ri,a,index_H1,3,ZZ,...
                  @f_HH,X0_HH,...
                   @F_H,X22_HH,...
                  @phi_HH,X22_HH,rc_1,...
                  1,XX,@f_PdPd,X0_PdPd,...
                   @phi_PdH,X4_PdH,rc_1,...
                   2,YY,@f_PdPd,X0_AgAg,...
                   @phi_PdH,X4_AgH,rc_1);

     %  H Central
    Ec_H_22 =    Ec3(ri,a,index_H2,3,ZZ,...
                  @f_HH,X0_HH,...
                   @F_H,X22_HH,...
                  @phi_HH,X22_HH,rc_1,...
                  1,XX,@f_PdPd,X0_PdPd,...
                   @phi_PdH,X4_PdH,rc_1,...
                   2,YY,@f_PdPd,X0_AgAg,...
                   @phi_PdH,X4_AgH,rc_1);


    Ec_Pd075Ag025H050_O1_O2   = ( xx * Ec_Pd_12 + xx * Ec_Pd_22 + xx *
Ec_Pd_32 + yy * Ec_Ag_12 + zz * Ec_H_12 + zz * Ec_H_22)/(3*xx + yy + 2*zz)


    Ec_Pd075Ag025H050_O1_O2_O1central   = ( 3*xx * Ec_Pd_12  + yy * Ec_Ag_12
+ 2*zz*Ec_H_12)/(3*xx + yy + 2*zz)
    Ec_Pd075Ag025H050_O1_O2_O2central   = ( 3*xx * Ec_Pd_12  + yy * Ec_Ag_12
+ 2*zz*Ec_H_22)/(3*xx + yy + 2*zz)


%************************************************************************
% For Pd075Ag025H025_O2_O2
```

```
%*************************************************************************
    global ri_PdAgH050_O2_O2;
    global index_PdAgH050_O2_O2_Pd1
    global index_PdAgH050_O2_O2_Pd2
    global index_PdAgH050_O2_O2_Pd3
    global index_PdAgH050_O2_O2_Ag
    global index_PdAgH050_O2_O2_H1
    global index_PdAgH050_O2_O2_H3

    ri = ri_PdAgH050_O2_O2;
    index_Pd1 = index_PdAgH050_O2_O2_Pd1;
    index_Pd2 = index_PdAgH050_O2_O2_Pd2;
    index_Pd3 = index_PdAgH050_O2_O2_Pd3;
    index_Ag = index_PdAgH050_O2_O2_Ag;
    index_H1  = index_PdAgH050_O2_O2_H1;
    index_H3  = index_PdAgH050_O2_O2_H3;

    global a_PdAgH050_O2_O2;
    a = a_PdAgH050_O2_O2;


    XX = 1;
    YY = 1;
    ZZ = 1;

    xx = 0.25;
    yy = 0.25;
    zz = 0.25;

    % Pd Central
    Ec_Pd_12 =    Ec3(ri,a,index_Pd1,1,XX,...
                    @f_PdPd,X0_PdPd,...
                    @F_Pd,X1_PdPd,...
                    @phi_PdPd,X2_PdPd,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_PdH,rc_1);

    % Pd Central
    Ec_Pd_22 =    Ec3(ri,a,index_Pd2,1,XX,...
                    @f_PdPd,X0_PdPd,...
                    @F_Pd,X1_PdPd,...
                    @phi_PdPd,X2_PdPd,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_PdH,rc_1);
    % Pd Central
    Ec_Pd_32 =    Ec3(ri,a,index_Pd3,1,XX,...
                    @f_PdPd,X0_PdPd,...
                    @F_Pd,X1_PdPd,...
                    @phi_PdPd,X2_PdPd,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdAg,X5_PdAg,rc_1,...
```

```
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_PdH,rc_1);

    % Ag Central
    Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
                   @f_PdPd,X0_AgAg,...
                    @F_Pd,X1_AgAg,...
                   @phi_PdPd,X2_AgAg,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_AgH,rc_1);

     % H Central
    Ec_H_12 =    Ec3(ri,a,index_H1,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

     % H Central
    Ec_H_22 =    Ec3(ri,a,index_H3,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);


    Ec_Pd075Ag025H050_O2_O2   = ( xx * Ec_Pd_12 + xx * Ec_Pd_22 + xx *
Ec_Pd_32 + yy * Ec_Ag_12 + zz * Ec_H_12 + zz * Ec_H_12)/(3*xx + yy + 2*zz)

%**************************************************************************
% For Pd075Ag025H050_TE1_TE2
%**************************************************************************

    global ri_PdAgH050_TE1_TE2;
    global index_PdAgH050_TE1_TE2_Pd1
    global index_PdAgH050_TE1_TE2_Pd2
    global index_PdAgH050_TE1_TE2_Pd3
    global index_PdAgH050_TE1_TE2_Ag
    global index_PdAgH050_TE1_TE2_H_TE1
    global index_PdAgH050_TE1_TE2_H_TE2

    ri = ri_PdAgH050_TE1_TE2;
    index_Pd1 = index_PdAgH050_TE1_TE2_Pd1;
    index_Pd2 = index_PdAgH050_TE1_TE2_Pd2;
    index_Pd3 = index_PdAgH050_TE1_TE2_Pd3;
    index_Ag = index_PdAgH050_TE1_TE2_Ag;
    index_H_TE1  = index_PdAgH050_TE1_TE2_H_TE1;
```

```
index_H_TE2  = index_PdAgH050_TE1_TE2_H_TE2;


global a_PdAgH050_TE1_TE2
a = a_PdAgH050_TE1_TE2;

XX = 1;
YY = 1;
ZZ = 1;

xx = 0.25;
yy = 0.25;
zz = 0.25;

% Pd Central
Ec_Pd_12 =    Ec3(ri,a,index_Pd1,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
% Pd Central
Ec_Pd_22 =    Ec3(ri,a,index_Pd2,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
% Pd Central
Ec_Pd_32 =    Ec3(ri,a,index_Pd3,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);

% Ag Central
Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
               @f_PdPd,X0_AgAg,...
                @F_Pd,X1_AgAg,...
               @phi_PdPd,X2_AgAg,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_AgH,rc_1);

 % H Central
Ec_H_12 =    Ec3(ri,a,index_H_TE1,3,ZZ,...
               @f_HH,X0_HH,...
```

```
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

     % H Central
    Ec_H_22 =    Ec3(ri,a,index_H_TE2,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

    Ec_Pd075Ag025H050_TE1_TE2   = ( xx*Ec_Pd_12 + xx*Ec_Pd_22 + xx*Ec_Pd_32 +
yy * Ec_Ag_12 + zz * Ec_H_12 + zz * Ec_H_22)/(3*xx + yy + 2*zz)


%************************************************************************
% For Pd075Ag025H050_TE1_TE8
%************************************************************************

    global ri_PdAgH050_TE1_TE8;
    global index_PdAgH050_TE1_TE8_Pd
    global index_PdAgH050_TE1_TE8_Ag
    global index_PdAgH050_TE1_TE8_H_TE1
    global index_PdAgH050_TE1_TE8_H_TE8


    ri = ri_PdAgH050_TE1_TE8;
    index_Pd = index_PdAgH050_TE1_TE8_Pd;
    index_Ag = index_PdAgH050_TE1_TE8_Ag;
    index_H_TE1  = index_PdAgH050_TE1_TE8_H_TE1;
    index_H_TE8  = index_PdAgH050_TE1_TE8_H_TE8;

    global a_PdAgH050_TE1_TE8
    a = a_PdAgH050_TE1_TE8;


    XX = 1;
    YY = 1;
    ZZ = 1;

    xx = 0.75;
    yy = 0.25;
    zz = 0.50;
    zzz = 0.25;

    % Pd Central
    Ec_Pd_12 =    Ec3(ri,a,index_Pd,1,XX,...
                    @f_PdPd,X0_PdPd,...
```

```
                    @F_Pd,X1_PdPd,...
                    @phi_PdPd,X2_PdPd,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_PdH,rc_1);
    % Ag Central
    Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
                   @f_PdPd,X0_AgAg,...
                    @F_Pd,X1_AgAg,...
                   @phi_PdPd,X2_AgAg,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_AgH,rc_1);

     % H Central
    Ec_H_12 =    Ec3(ri,a,index_H_TE1,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

    Ec_Pd075Ag025H050_TE1_TE8__TE1central   = ( xx * Ec_Pd_12  + yy *
Ec_Ag_12 + zz * Ec_H_12)/(xx + yy + zz)

     % H Central
    Ec_H_22 =    Ec3(ri,a,index_H_TE8,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

    Ec_Pd075Ag025H050_TE1_TE8   = ( xx * Ec_Pd_12  + yy * Ec_Ag_12 + zzz *
Ec_H_12+ zzz * Ec_H_22)/(xx + yy + zzz + zzz)



%************************************************************************
% For Pd075Ag025H050_TE3_TE4
%************************************************************************

    global ri_PdAgH050_TE3_TE4
    global index_PdAgH050_TE3_TE4_Pd
    global index_PdAgH050_TE3_TE4_Ag
    global index_PdAgH050_TE3_TE4_H_TE3
    global index_PdAgH050_TE3_TE4_H_TE4

    ri = ri_PdAgH050_TE3_TE4;
```

```
index_Pd = index_PdAgH050_TE3_TE4_Pd;
index_Ag = index_PdAgH050_TE3_TE4_Ag;
index_H_TE3  = index_PdAgH050_TE3_TE4_H_TE3;
index_H_TE4  = index_PdAgH050_TE3_TE4_H_TE4;

global a_PdAgH050_TE3_TE4
a = a_PdAgH050_TE3_TE4;

XX = 1;
YY = 1;
ZZ = 1;

xx = 0.75;
yy = 0.25;
zz = 0.25;
zzz = 0.25;

% Pd Central
Ec_Pd_12 =    Ec3(ri,a,index_Pd,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
% Ag Central
Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
               @f_PdPd,X0_AgAg,...
                @F_Pd,X1_AgAg,...
               @phi_PdPd,X2_AgAg,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_AgH,rc_1);

 % H Central
Ec_H_12 =    Ec3(ri,a,index_H_TE3,3,ZZ,...
               @f_HH,X0_HH,...
                @F_H,X22_HH,...
               @phi_HH,X22_HH,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdH,X4_PdH,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdH,X4_AgH,rc_1);

  % H Central
Ec_H_22 =    Ec3(ri,a,index_H_TE4,3,ZZ,...
               @f_HH,X0_HH,...
                @F_H,X22_HH,...
               @phi_HH,X22_HH,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdH,X4_PdH,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdH,X4_AgH,rc_1);
```

```
    Ec_Pd075Ag025H050_TE3_TE4   = ( xx * Ec_Pd_12  + yy * Ec_Ag_12 + zz *
Ec_H_12 + zzz * Ec_H_22)/(xx + yy + zz + zzz)


%**************************************************************************
% For Pd075Ag025H050_TE3_TE6
%**************************************************************************

    global ri_PdAgH050_TE3_TE6
    global index_PdAgH050_TE3_TE6_Pd
    global index_PdAgH050_TE3_TE6_Ag
    global index_PdAgH050_TE3_TE6_H_TE3
    global index_PdAgH050_TE3_TE6_H_TE6

    ri = ri_PdAgH050_TE3_TE6;
    index_Pd = index_PdAgH050_TE3_TE6_Pd;
    index_Ag = index_PdAgH050_TE3_TE6_Ag;
    index_H_TE3  = index_PdAgH050_TE3_TE6_H_TE3;
    index_H_TE6  = index_PdAgH050_TE3_TE6_H_TE6;

    global a_PdAgH050_TE3_TE6
    a = a_PdAgH050_TE3_TE6;

    XX = 1;
    YY = 1;
    ZZ = 1;

    xx = 0.75;
    yy = 0.25;
    zz = 0.25;
    zzz = 0.25;

    % Pd Central
    Ec_Pd_12 =    Ec3(ri,a,index_Pd,1,XX,...
                    @f_PdPd,X0_PdPd,...
                    @F_Pd,X1_PdPd,...
                    @phi_PdPd,X2_PdPd,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_PdH,rc_1);
    % Ag Central
    Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
                   @f_PdPd,X0_AgAg,...
                    @F_Pd,X1_AgAg,...
                   @phi_PdPd,X2_AgAg,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_AgH,rc_1);

     % H Central
    Ec_H_12 =    Ec3(ri,a,index_H_TE3,3,ZZ,...
```

```
                    @f_HH,X0_HH,...
                     @F_H,X22_HH,...
                    @phi_HH,X22_HH,rc_1,...
                    1,XX,@f_PdPd,X0_PdPd,...
                     @phi_PdH,X4_PdH,rc_1,...
                     2,YY,@f_PdPd,X0_AgAg,...
                     @phi_PdH,X4_AgH,rc_1);

      % H Central
    Ec_H_22 =    Ec3(ri,a,index_H_TE6,3,ZZ,...
                    @f_HH,X0_HH,...
                     @F_H,X22_HH,...
                    @phi_HH,X22_HH,rc_1,...
                    1,XX,@f_PdPd,X0_PdPd,...
                     @phi_PdH,X4_PdH,rc_1,...
                     2,YY,@f_PdPd,X0_AgAg,...
                     @phi_PdH,X4_AgH,rc_1);


    Ec_Pd075Ag025H050_TE3_TE6   = ( xx * Ec_Pd_12  + yy * Ec_Ag_12 + zz *
Ec_H_12 + zzz * Ec_H_22)/(xx + yy + zz + zzz)



%************************************************************************
% For Pd075Ag025H075_O1_O2_O2
%************************************************************************
    global ri_PdAgH075_O1_O2_O2;
    global index_PdAgH075_O1_O2_O2_Pd1
    global index_PdAgH075_O1_O2_O2_Pd2
    global index_PdAgH075_O1_O2_O2_Pd3
    global index_PdAgH075_O1_O2_O2_Ag
    global index_PdAgH075_O1_O2_O2_H1
    global index_PdAgH075_O1_O2_O2_H2
    global index_PdAgH075_O1_O2_O2_H3

    ri = ri_PdAgH075_O1_O2_O2;
    index_Pd1 = index_PdAgH075_O1_O2_O2_Pd1;
    index_Pd2 = index_PdAgH075_O1_O2_O2_Pd2;
    index_Pd3 = index_PdAgH075_O1_O2_O2_Pd3;
    index_Ag = index_PdAgH075_O1_O2_O2_Ag;
    index_H1  = index_PdAgH075_O1_O2_O2_H1;
    index_H2  = index_PdAgH075_O1_O2_O2_H2;
    index_H3  = index_PdAgH075_O1_O2_O2_H3;

    global a_PdAgH075_O1_O2_O2;
    a = a_PdAgH075_O1_O2_O2;

    XX = 1;
    YY = 1;
    ZZ = 1;

    xx = 0.25;
    yy = 0.25;
    zz = 0.25;
```

```
% Pd Central
Ec_Pd_12 =    Ec3(ri,a,index_Pd1,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
% Pd Central
Ec_Pd_22 =    Ec3(ri,a,index_Pd2,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
% Pd Central
Ec_Pd_32 =    Ec3(ri,a,index_Pd3,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);

 % Ag Central
Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
               @f_PdPd,X0_AgAg,...
                @F_Pd,X1_AgAg,...
               @phi_PdPd,X2_AgAg,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_AgH,rc_1);

 %  H Central
Ec_H_12 =    Ec3(ri,a,index_H1,3,ZZ,...
               @f_HH,X0_HH,...
                @F_H,X22_HH,...
               @phi_HH,X22_HH,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdH,X4_PdH,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdH,X4_AgH,rc_1);

  %  H Central
Ec_H_22 =    Ec3(ri,a,index_H2,3,ZZ,...
               @f_HH,X0_HH,...
                @F_H,X22_HH,...
               @phi_HH,X22_HH,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
```

```
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

      %  H Central
    Ec_H_32 =    Ec3(ri,a,index_H3,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

    Ec_Pd075Ag025H075_O1_O2_O2   = ( xx*Ec_Pd_12 + xx*Ec_Pd_22 + xx*Ec_Pd_32
+ yy * Ec_Ag_12 + zz * Ec_H_12 + zz * Ec_H_22 + zz * Ec_H_32)/(3*xx + yy +
3*zz)


    Ec_Pd075Ag025H075_O1_O2_O2_O1central   = ( 3*xx*Ec_Pd_12 + yy * Ec_Ag_12
+ 3*zz*Ec_H_12)/(3*xx + yy + 3*zz)

    Ec_Pd075Ag025H075_O1_O2_O2_O2central   = ( 3*xx*Ec_Pd_12 + yy * Ec_Ag_12
+ 3*zz*Ec_H_22)/(3*xx + yy + 3*zz)


%**************************************************************************
% For Pd075Ag025H075_O2_O2_O2
%**************************************************************************
    global ri_PdAgH075_O2_O2_O2;
    global index_PdAgH075_O2_O2_O2_Pd1
    global index_PdAgH075_O2_O2_O2_Pd2
    global index_PdAgH075_O2_O2_O2_Pd3
    global index_PdAgH075_O2_O2_O2_Ag
    global index_PdAgH075_O2_O2_O2_H1
    global index_PdAgH075_O2_O2_O2_H2
    global index_PdAgH075_O2_O2_O2_H3

    ri = ri_PdAgH075_O2_O2_O2;
    index_Pd = index_PdAgH075_O2_O2_O2_Pd1;
    index_Pd = index_PdAgH075_O2_O2_O2_Pd2;
    index_Pd = index_PdAgH075_O2_O2_O2_Pd3;
    index_Ag = index_PdAgH075_O2_O2_O2_Ag;
    index_H  = index_PdAgH075_O2_O2_O2_H1;
    index_H  = index_PdAgH075_O2_O2_O2_H2;
    index_H  = index_PdAgH075_O2_O2_O2_H3;

    global a_PdAgH075_O2_O2_O2;
    a = a_PdAgH075_O2_O2_O2;

    XX = 1;
    YY = 1;
    ZZ = 1;

    xx = 0.25;
```

```
yy = 0.25;
zz = 0.25;

% Pd Central
Ec_Pd_12 =    Ec3(ri,a,index_Pd1,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
% Pd Central
Ec_Pd_22 =    Ec3(ri,a,index_Pd2,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
% Pd Central
Ec_Pd_32 =    Ec3(ri,a,index_Pd3,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);

 % Ag Central
Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
               @f_PdPd,X0_AgAg,...
                @F_Pd,X1_AgAg,...
               @phi_PdPd,X2_AgAg,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_AgH,rc_1);

 %  H Central
Ec_H_12 =    Ec3(ri,a,index_H1,3,ZZ,...
               @f_HH,X0_HH,...
                @F_H,X22_HH,...
               @phi_HH,X22_HH,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdH,X4_PdH,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdH,X4_AgH,rc_1);
 %  H Central
Ec_H_22 =    Ec3(ri,a,index_H2,3,ZZ,...
               @f_HH,X0_HH,...
                @F_H,X22_HH,...
               @phi_HH,X22_HH,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
```

```
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);
     %  H Central
    Ec_H_32 =    Ec3(ri,a,index_H3,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

    Ec_Pd075Ag025H075_O2_O2_O2   = ( xx * Ec_Pd_12 + xx * Ec_Pd_22 + xx *
Ec_Pd_32 + yy * Ec_Ag_12 + zz * Ec_H_12 + zz * Ec_H_22 + zz * Ec_H_32)/(3*xx
+ yy + 3*zz)


%************************************************************************
% For Pd075Ag025H075_T
%************************************************************************
    global ri_PdAgH075_T;
    global index_PdAgH075_T_Pd
    global index_PdAgH075_T_Ag
    global index_PdAgH075_T_H

    ri = ri_PdAgH075_T;
    index_Pd = index_PdAgH075_T_Pd;
    index_Ag = index_PdAgH075_T_Ag;
    index_H  = index_PdAgH075_T_H;

    global a_PdAgH075_TE5_TE6_TE7
    a = a_PdAgH075_TE5_TE6_TE7;

    XX = 1;
    YY = 1;
    ZZ = 1;

    xx = 0.75;
    yy = 0.25;
    zz = 0.75;

    % Pd Central
    Ec_Pd_12 =    Ec3(ri,a,index_Pd,1,XX,...
                    @f_PdPd,X0_PdPd,...
                    @F_Pd,X1_PdPd,...
                    @phi_PdPd,X2_PdPd,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_PdH,rc_1);
    % Ag Central
    Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
                   @f_PdPd,X0_AgAg,...
                    @F_Pd,X1_AgAg,...
```

```
                    @phi_PdPd,X2_AgAg,rc_1,...
                    1,XX,@f_PdPd,X0_PdPd,...
                     @phi_PdAg,X5_PdAg,rc_1,...
                     3,ZZ,@f_HH,X0_HH,...
                     @phi_PdH,X4_AgH,rc_1);

     %  H Central
    Ec_H_12 =    Ec3(ri,a,index_H,3,ZZ,...
                    @f_HH,X0_HH,...
                     @F_H,X22_HH,...
                    @phi_HH,X22_HH,rc_1,...
                    1,XX,@f_PdPd,X0_PdPd,...
                     @phi_PdH,X4_PdH,rc_1,...
                     2,YY,@f_PdPd,X0_AgAg,...
                     @phi_PdH,X4_AgH,rc_1);



    Ec_Pd075Ag025H075_TE5_TE6_TE7   = ( xx * Ec_Pd_12  + yy * Ec_Ag_12 + zz *
Ec_H_12)/(xx + yy + zz)

%************************************************************************
% For Pd075Ag025H075_TE1_TE2_TE3
%************************************************************************

    global ri_PdAgH075_TE1_TE2_TE3
    global index_PdAgH075_TE1_TE2_TE3_Pd1
    global index_PdAgH075_TE1_TE2_TE3_Pd2
    global index_PdAgH075_TE1_TE2_TE3_Pd3
    global index_PdAgH075_TE1_TE2_TE3_Ag
    global index_PdAgH075_TE1_TE2_TE3_H_TE1
    global index_PdAgH075_TE1_TE2_TE3_H_TE2
    global index_PdAgH075_TE1_TE2_TE3_H_TE3

    ri = ri_PdAgH075_TE1_TE2_TE3;
    index_Pd1 = index_PdAgH075_TE1_TE2_TE3_Pd1;
    index_Pd2 = index_PdAgH075_TE1_TE2_TE3_Pd2;
    index_Pd3 = index_PdAgH075_TE1_TE2_TE3_Pd3;
    index_Ag = index_PdAgH075_TE1_TE2_TE3_Ag;
    index_H_TE1  = index_PdAgH075_TE1_TE2_TE3_H_TE1;
    index_H_TE2  = index_PdAgH075_TE1_TE2_TE3_H_TE2;
    index_H_TE3  = index_PdAgH075_TE1_TE2_TE3_H_TE3;

    global a_PdAgH075_TE1_TE2_TE3
    a = a_PdAgH075_TE1_TE2_TE3;

    XX = 1;
    YY = 1;
    ZZ = 1;

    xx = 0.25;
    yy = 0.25;
    zz = 0.25;
```

```
% Pd Central
Ec_Pd_12 =    Ec3(ri,a,index_Pd1,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
% Pd Central
Ec_Pd_22 =    Ec3(ri,a,index_Pd2,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
% Pd Central
Ec_Pd_32 =    Ec3(ri,a,index_Pd3,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);

% Ag Central
Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
               @f_PdPd,X0_AgAg,...
                @F_Pd,X1_AgAg,...
               @phi_PdPd,X2_AgAg,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_AgH,rc_1);

 %  H Central
Ec_H_12 =    Ec3(ri,a,index_H_TE1,3,ZZ,...
               @f_HH,X0_HH,...
                @F_H,X22_HH,...
               @phi_HH,X22_HH,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdH,X4_PdH,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdH,X4_AgH,rc_1);

 %  H Central
Ec_H_22 =    Ec3(ri,a,index_H_TE2,3,ZZ,...
               @f_HH,X0_HH,...
                @F_H,X22_HH,...
               @phi_HH,X22_HH,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdH,X4_PdH,rc_1,...
```

```
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

     %  H Central
    Ec_H_32 =    Ec3(ri,a,index_H_TE3,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

    Ec_Pd075Ag025H075_TE1_TE2_TE3   = ( xx*Ec_Pd_12 + xx*Ec_Pd_22 +
xx*Ec_Pd_32 + yy * Ec_Ag_12 + zz * Ec_H_12 + zz * Ec_H_22 + zz *
Ec_H_32)/(3*xx + yy + 3*zz)


%************************************************************************
% For Pd075Ag025H075_TE1_TE2_TE8
%************************************************************************

    global ri_PdAgH075_TE1_TE2_TE8
    global index_PdAgH075_TE1_TE2_TE8_Pd1
    global index_PdAgH075_TE1_TE2_TE8_Pd2
    global index_PdAgH075_TE1_TE2_TE8_Pd3
    global index_PdAgH075_TE1_TE2_TE8_Ag
    global index_PdAgH075_TE1_TE2_TE8_H_TE1
    global index_PdAgH075_TE1_TE2_TE8_H_TE2
    global index_PdAgH075_TE1_TE2_TE8_H_TE8

    ri = ri_PdAgH075_TE1_TE2_TE8;
    index_Pd1 = index_PdAgH075_TE1_TE2_TE8_Pd1;
    index_Pd2 = index_PdAgH075_TE1_TE2_TE8_Pd2;
    index_Pd3 = index_PdAgH075_TE1_TE2_TE8_Pd3;
    index_Ag = index_PdAgH075_TE1_TE2_TE8_Ag;
    index_H_TE1  = index_PdAgH075_TE1_TE2_TE8_H_TE1;
    index_H_TE2  = index_PdAgH075_TE1_TE2_TE8_H_TE2;
    index_H_TE8  = index_PdAgH075_TE1_TE2_TE8_H_TE8;

    global a_PdAgH075_TE1_TE2_TE8
    a = a_PdAgH075_TE1_TE2_TE8;

    XX = 1;
    YY = 1;
    ZZ = 1;

    xx = 0.25;
    yy = 0.25;
    zz = 0.25;

    % Pd Central
    Ec_Pd_12 =    Ec3(ri,a,index_Pd1,1,XX,...
                    @f_PdPd,X0_PdPd,...
                    @F_Pd,X1_PdPd,...
```

```
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
% Pd Central
Ec_Pd_22 =    Ec3(ri,a,index_Pd2,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
% Pd Central
Ec_Pd_32 =    Ec3(ri,a,index_Pd3,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);

% Ag Central
Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
               @f_PdPd,X0_AgAg,...
                @F_Pd,X1_AgAg,...
               @phi_PdPd,X2_AgAg,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_AgH,rc_1);

 %  H Central
Ec_H_12 =    Ec3(ri,a,index_H_TE1,3,ZZ,...
               @f_HH,X0_HH,...
                @F_H,X22_HH,...
               @phi_HH,X22_HH,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdH,X4_PdH,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdH,X4_AgH,rc_1);

 %  H Central
Ec_H_22 =    Ec3(ri,a,index_H_TE2,3,ZZ,...
               @f_HH,X0_HH,...
                @F_H,X22_HH,...
               @phi_HH,X22_HH,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdH,X4_PdH,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdH,X4_AgH,rc_1);

 %  H Central
Ec_H_32 =    Ec3(ri,a,index_H_TE8,3,ZZ,...
```

```
                    @f_HH,X0_HH,...
                     @F_H,X22_HH,...
                    @phi_HH,X22_HH,rc_1,...
                    1,XX,@f_PdPd,X0_PdPd,...
                     @phi_PdH,X4_PdH,rc_1,...
                     2,YY,@f_PdPd,X0_AgAg,...
                     @phi_PdH,X4_AgH,rc_1);

    Ec_Pd075Ag025H075_TE1_TE2_TE8   = ( xx*Ec_Pd_12 + xx*Ec_Pd_22 +
xx*Ec_Pd_32 + yy * Ec_Ag_12 + zz * Ec_H_12 + zz * Ec_H_22 + zz *
Ec_H_32)/(3*xx + yy + 3*zz)


%************************************************************************
% For Pd075Ag025H100_O1_O2_O2_O2
%************************************************************************
    global ri_PdAgH100_O1_O2_O2_O2;
    global index_PdAgH100_O1_O2_O2_O2_Pd1
    global index_PdAgH100_O1_O2_O2_O2_Pd2
    global index_PdAgH100_O1_O2_O2_O2_Pd3
    global index_PdAgH100_O1_O2_O2_O2_Ag
    global index_PdAgH100_O1_O2_O2_O2_H1
    global index_PdAgH100_O1_O2_O2_O2_H2
    global index_PdAgH100_O1_O2_O2_O2_H3
    global index_PdAgH100_O1_O2_O2_O2_H4


    ri = ri_PdAgH100_O1_O2_O2_O2;
    index_Pd1 = index_PdAgH100_O1_O2_O2_O2_Pd1;
    index_Pd2 = index_PdAgH100_O1_O2_O2_O2_Pd2;
    index_Pd3 = index_PdAgH100_O1_O2_O2_O2_Pd3;
    index_Ag = index_PdAgH100_O1_O2_O2_O2_Ag;
    index_H1  = index_PdAgH100_O1_O2_O2_O2_H1;
    index_H2  = index_PdAgH100_O1_O2_O2_O2_H2;
    index_H3  = index_PdAgH100_O1_O2_O2_O2_H3;
    index_H4  = index_PdAgH100_O1_O2_O2_O2_H4;

    global a_PdAgH100_O1_O2_O2_O2;
    a = a_PdAgH100_O1_O2_O2_O2;

    XX = 1;
    YY = 1;
    ZZ = 1;

    xx = 0.25;
    yy = 0.25;
    zz = 0.25;

    % Pd Central
    Ec_Pd_12 =    Ec3(ri,a,index_Pd1,1,XX,...
                    @f_PdPd,X0_PdPd,...
                    @F_Pd,X1_PdPd,...
                    @phi_PdPd,X2_PdPd,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdAg,X5_PdAg,rc_1,...
```

```
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);

% Pd Central
Ec_Pd_22 =    Ec3(ri,a,index_Pd2,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);

% Pd Central
Ec_Pd_32 =    Ec3(ri,a,index_Pd3,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);

 % Ag Central
Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
               @f_PdPd,X0_AgAg,...
                @F_Pd,X1_AgAg,...
               @phi_PdPd,X2_AgAg,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_AgH,rc_1);

 %  H Central
Ec_H_12 =    Ec3(ri,a,index_H1,3,ZZ,...
               @f_HH,X0_HH,...
                @F_H,X22_HH,...
               @phi_HH,X22_HH,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdH,X4_PdH,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdH,X4_AgH,rc_1);


 %  H Central
Ec_H_22 =    Ec3(ri,a,index_H2,3,ZZ,...
               @f_HH,X0_HH,...
                @F_H,X22_HH,...
               @phi_HH,X22_HH,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdH,X4_PdH,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdH,X4_AgH,rc_1);

 %  H Central
```

```
    Ec_H_32 =    Ec3(ri,a,index_H3,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

     %  H Central
    Ec_H_42 =    Ec3(ri,a,index_H4,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);


    Ec_Pd075Ag025H100_O1_O2_O2_O2   = ( xx * Ec_Pd_12 + xx * Ec_Pd_22 + xx *
Ec_Pd_32 + yy * Ec_Ag_12 + zz * Ec_H_12 + zz * Ec_H_22 + zz * Ec_H_32 + zz *
Ec_H_42)/(3*xx + yy + 4*zz)

    Ec_Pd075Ag025H100_O1_O2_O2_O2_O1central  =  ( 3*xx*Ec_Pd_12  +
yy*Ec_Ag_12 + 4*zz*Ec_H_12)/(3*xx + yy + 4*zz)

    Ec_Pd075Ag025H100_O1_O2_O2_O2_O2central   = ( 3*xx*Ec_Pd_12  +
yy*Ec_Ag_12 + 4*zz*Ec_H_22)/(3*xx + yy + 4*zz)


%************************************************************************
% For Pd075Ag025H100_T
%************************************************************************
    global ri_PdAgH100_T;
    global index_PdAgH100_T_Pd
    global index_PdAgH100_T_Ag
    global index_PdAgH100_T_H

    ri = ri_PdAgH100_T;
    index_Pd = index_PdAgH100_T_Pd;
    index_Ag = index_PdAgH100_T_Ag;
    index_H  = index_PdAgH100_T_H;

    global a_PdAgH100_TE1_TE5_TE6_TE7
    a = a_PdAgH100_TE1_TE5_TE6_TE7;

    XX = 1;
    YY = 1;
    ZZ = 1;

    xx = 0.75;
    yy = 0.25;
    zz = 1.00;
```

```
    % Pd Central
    Ec_Pd_12 =    Ec3(ri,a,index_Pd,1,XX,...
                    @f_PdPd,X0_PdPd,...
                    @F_Pd,X1_PdPd,...
                    @phi_PdPd,X2_PdPd,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_PdH,rc_1);
    % Ag Central
    Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
                   @f_PdPd,X0_AgAg,...
                    @F_Pd,X1_AgAg,...
                   @phi_PdPd,X2_AgAg,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdAg,X5_PdAg,rc_1,...
                    3,ZZ,@f_HH,X0_HH,...
                    @phi_PdH,X4_AgH,rc_1);

     %  H Central
    Ec_H_12 =    Ec3(ri,a,index_H,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);



    Ec_Pd075Ag025H100_TE1_TE5_TE6_TE7   = ( xx * Ec_Pd_12  + yy * Ec_Ag_12 +
zz * Ec_H_12)/(xx + yy + zz)

%**************************************************************************
% For Pd075Ag025H100_TE1_TE2_TE3_TE4
%**************************************************************************

    global ri_PdAgH100_TE1_TE2_TE3_TE4
    global index_PdAgH100_TE1_TE2_TE3_TE4_Pd1
    global index_PdAgH100_TE1_TE2_TE3_TE4_Pd2
    global index_PdAgH100_TE1_TE2_TE3_TE4_Pd3
    global index_PdAgH100_TE1_TE2_TE3_TE4_Ag
    global index_PdAgH100_TE1_TE2_TE3_TE4_H_TE1
    global index_PdAgH100_TE1_TE2_TE3_TE4_H_TE2
    global index_PdAgH100_TE1_TE2_TE3_TE4_H_TE3
    global index_PdAgH100_TE1_TE2_TE3_TE4_H_TE4

    ri = ri_PdAgH100_TE1_TE2_TE3_TE4;
    index_Pd1 = index_PdAgH100_TE1_TE2_TE3_TE4_Pd1;
    index_Pd2 = index_PdAgH100_TE1_TE2_TE3_TE4_Pd2;
    index_Pd3 = index_PdAgH100_TE1_TE2_TE3_TE4_Pd3;
    index_Ag = index_PdAgH100_TE1_TE2_TE3_TE4_Ag;
    index_H_TE1  = index_PdAgH100_TE1_TE2_TE3_TE4_H_TE1;
```

```
index_H_TE2  = index_PdAgH100_TE1_TE2_TE3_TE4_H_TE2;
index_H_TE3  = index_PdAgH100_TE1_TE2_TE3_TE4_H_TE3;
index_H_TE4  = index_PdAgH100_TE1_TE2_TE3_TE4_H_TE4;

global a_PdAgH100_TE1_TE2_TE3_TE4
a = a_PdAgH100_TE1_TE2_TE3_TE4;

XX = 1;
YY = 1;
ZZ = 1;

xx = 0.25;
yy = 0.25;
zz = 0.25;

% Pd Central
Ec_Pd_12 =    Ec3(ri,a,index_Pd1,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
% Pd Central
Ec_Pd_22 =    Ec3(ri,a,index_Pd2,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
% Pd Central
Ec_Pd_32 =    Ec3(ri,a,index_Pd3,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);

% Ag Central
Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
               @f_PdPd,X0_AgAg,...
                @F_Pd,X1_AgAg,...
               @phi_PdPd,X2_AgAg,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_AgH,rc_1);

 %  H Central
Ec_H_12 =    Ec3(ri,a,index_H_TE1,3,ZZ,...
               @f_HH,X0_HH,...
```

```
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

     %  H Central
    Ec_H_22 =    Ec3(ri,a,index_H_TE2,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

     %  H Central
    Ec_H_32 =    Ec3(ri,a,index_H_TE3,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

     %  H Central
    Ec_H_42 =    Ec3(ri,a,index_H_TE4,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);


    Ec_Pd075Ag025H100_TE1_TE2_TE3_TE4   = ( xx*Ec_Pd_12 + xx*Ec_Pd_22 +
xx*Ec_Pd_32 + yy * Ec_Ag_12 + zz * Ec_H_12 + zz * Ec_H_22 + zz * Ec_H_32 + zz
* Ec_H_42)/(3*xx + yy + 4*zz)


%************************************************************************
% For Pd075Ag025H100_TE1_TE2_TE7_TE8
%************************************************************************

    global ri_PdAgH100_TE1_TE2_TE7_TE8
    global index_PdAgH100_TE1_TE2_TE7_TE8_Pd
    global index_PdAgH100_TE1_TE2_TE7_TE8_Ag
    global index_PdAgH100_TE1_TE2_TE7_TE8_H_TE1
    global index_PdAgH100_TE1_TE2_TE7_TE8_H_TE2
    global index_PdAgH100_TE1_TE2_TE7_TE8_H_TE7
    global index_PdAgH100_TE1_TE2_TE7_TE8_H_TE8
```

```
ri = ri_PdAgH100_TE1_TE2_TE7_TE8;
index_Pd = index_PdAgH100_TE1_TE2_TE7_TE8_Pd;
index_Ag = index_PdAgH100_TE1_TE2_TE7_TE8_Ag;
index_H_TE1  = index_PdAgH100_TE1_TE2_TE7_TE8_H_TE1;
index_H_TE2  = index_PdAgH100_TE1_TE2_TE7_TE8_H_TE2;
index_H_TE7  = index_PdAgH100_TE1_TE2_TE7_TE8_H_TE7;
index_H_TE8  = index_PdAgH100_TE1_TE2_TE7_TE8_H_TE8;


global a_PdAgH100_TE1_TE2_TE7_TE8
a = a_PdAgH100_TE1_TE2_TE7_TE8;

XX = 1;
YY = 1;
ZZ = 1;

xx = 0.75;
yy = 0.25;
zz = 0.25;

% Pd Central
Ec_Pd_12 =    Ec3(ri,a,index_Pd,1,XX,...
                @f_PdPd,X0_PdPd,...
                @F_Pd,X1_PdPd,...
                @phi_PdPd,X2_PdPd,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_PdH,rc_1);
% Ag Central
Ec_Ag_12 =    Ec3(ri,a,index_Ag,2,YY,...
               @f_PdPd,X0_AgAg,...
                @F_Pd,X1_AgAg,...
               @phi_PdPd,X2_AgAg,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdAg,X5_PdAg,rc_1,...
                3,ZZ,@f_HH,X0_HH,...
                @phi_PdH,X4_AgH,rc_1);

 %  H Central
Ec_H_12 =    Ec3(ri,a,index_H_TE1,3,ZZ,...
               @f_HH,X0_HH,...
                @F_H,X22_HH,...
               @phi_HH,X22_HH,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
                @phi_PdH,X4_PdH,rc_1,...
                2,YY,@f_PdPd,X0_AgAg,...
                @phi_PdH,X4_AgH,rc_1);

 %  H Central
Ec_H_22 =    Ec3(ri,a,index_H_TE2,3,ZZ,...
               @f_HH,X0_HH,...
                @F_H,X22_HH,...
               @phi_HH,X22_HH,rc_1,...
               1,XX,@f_PdPd,X0_PdPd,...
```

```
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

     %  H Central
    Ec_H_32 =    Ec3(ri,a,index_H_TE7,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);

     %  H Central
    Ec_H_42 =    Ec3(ri,a,index_H_TE8,3,ZZ,...
                   @f_HH,X0_HH,...
                    @F_H,X22_HH,...
                   @phi_HH,X22_HH,rc_1,...
                   1,XX,@f_PdPd,X0_PdPd,...
                    @phi_PdH,X4_PdH,rc_1,...
                    2,YY,@f_PdPd,X0_AgAg,...
                    @phi_PdH,X4_AgH,rc_1);


    Ec_Pd075Ag025H100_TE1_TE2_TE7_TE8   = ( xx * Ec_Pd_12  + yy * Ec_Ag_12 +
zz * Ec_H_12 + zz * Ec_H_22 + zz * Ec_H_32 + zz * Ec_H_42)/(xx + yy + 4*zz)



    %************************************************************************
    % Table
    %************************************************************************


    Model        = [Ec_Pd075Ag025H025_O1; Ec_Pd075Ag025H025_O2;
Ec_Pd075Ag025H025_TE1;...

Ec_Pd075Ag025H050_O1_O2;Ec_Pd075Ag025H050_O2_O2;Ec_Pd075Ag025H050_TE1_TE2;...

Ec_Pd075Ag025H050_TE1_TE8;Ec_Pd075Ag025H075_O1_O2_O2;Ec_Pd075Ag025H075_O2_O2_
O2;...

Ec_Pd075Ag025H075_TE5_TE6_TE7;Ec_Pd075Ag025H100_O1_O2_O2_O2;Ec_Pd075Ag025H100
_TE1_TE5_TE6_TE7];

    Siesta = [Ec_exp_PdAgH025_O1; Ec_exp_PdAgH025_O2; Ec_exp_PdAgH025_T;...
           Ec_exp_PdAgH050_O1_O2; Ec_exp_PdAgH050_O2_O2; Ec_exp_PdAgH050_T;
Ec_exp_PdAgH050_T;...
           Ec_exp_PdAgH075_O1_O2_O2; Ec_exp_PdAgH075_O2_O2_O2;
Ec_exp_PdAgH075_T;...
           Ec_exp_PdAgH100_O1_O2_O2_O2; Ec_exp_PdAgH100_T];
```

```
    Zhou_Experimental = [-3.396;-3.337;-3.373;-3.180;-3.147;-3.171;-3.171;-
3.035;-2.944;-3.025;-2.863;-2.919];

    Zhou = [-3.49499221;-3.2892551;-3.4706926;-3.23099;-3.086482;-3.168897;-
3.168897;-3.070289;-2.9575078;-3.031653827;-2.95657414;-3.0171908];

%     PError       = abs(Model-Experimental).*100./abs(Experimental);

    Rownames =
{'PdAgH025_O1','PdAgH025_O2','PdAgH025_TE1','PdAgH050_O1_O2',...

'PdAgH050_O2_O2','PdAgH050_TE1_TE2','PdAgH050_TE1_TE8','PdAgH075_O1_O2_O2',..
.

'PdAgH075_O2_O2_O2','PdAgH075_TE5_TE6_TE7','PdAgH100_O1_O2_O2_O2','PdAgH100_T
E1_TE5_TE6_TE7'};

    Columnames = {'Siesta';'Model';'Zhou_MD';'Zhou_DFT'};

    Results =
table(Siesta,Model,Zhou,Zhou_Experimental,'VariableNames',Columnames,'RowName
s',Rownames)



    ar = 2.1075;
    fig_height = 6;
    fig_width = ar*fig_height;


     %  Two Body Potential
    figure(1)
    j = 1;
    for r = 0.4:0.01:6
       Phi_AgH(j) = phi_PdH(r,X4_AgH);
       Phi_AgH_Zhou_Morse(j) = phi_PdH(r,X4_AgH_Zhou_Morse);
       Phi_AgH_Zhou_mix(j) = phi_PdH(r,X4_AgH_Zhou_mix);
       j = j + 1;
    end
    r = [0.4:0.01:6];
    plot(r,Phi_AgH,'black-
',r,Phi_AgH_Zhou_Morse,'black:',r,Phi_AgH_Zhou_mix,'black-.','markersize',3,
'linewidth',4)
    xlabel('r (A^o)','fontsize',24)
    ylabel('Two Body Potential (eV)','fontsize',24)
    legend('AgH','Zhou Morse','Zhou Mixing')
    set(gca,'FontSize',24,'FontWeight','bold','linewidth',3)
    axis([0.0,6,-1,2.5])
    fig = figure(1);
    fig.Units = 'inches';
    fig.Position = [2 2 fig_width fig_height];
```

```
     % Comparison Two Body Potential
    figure(2)
    j = 1;
    for r = 0.4:0.01:6
       Phi_PdPd(j)    = phi_PdPd(r,X2_PdPd);
       Phi_AgAg(j)    = phi_PdPd(r,X2_AgAg);
       Phi_HH(j)  = phi_HH(r,X22_HH);
       Phi_PdH(j) = phi_PdH(r,X4_PdH);
       Phi_AgH(j) = phi_PdH(r,X4_AgH);
       Phi_PdAg(j) = phi_PdAg(r,X5_PdAg);
       j = j + 1;
    end
    r = [0.4:0.01:6];
    plot(r,Phi_HH,'black:',r,Phi_PdH,'black-.',r,Phi_PdPd,'black--
',r,Phi_AgAg,'g:',r,Phi_AgH,'black-',r,Phi_PdAg,'b:','markersize',3,
'linewidth',4)
    xlabel('r (A^o)','fontsize',24)
    ylabel('Two Body Potential (eV)','fontsize',24)
    legend('HH','PdH','PdPd','AgAg','AgH','PdAg')
    set(gca,'FontSize',24,'FontWeight','bold','linewidth',3)
    axis([0.0,6,-1,2.5])
    fig = figure(2);
    fig.Units = 'inches';
    fig.Position = [2 2 fig_width fig_height];

     % Comparison Two Body Potential
    figure(22)
    plot(r,Phi_HH,'k:',r,Phi_PdH,'k-.',r,Phi_AgH,'k--','markersize',3,
'linewidth',4)
    xlabel('r (A^o)','fontsize',24)
    ylabel('Two Body Potential (eV)','fontsize',24)
    legend('HH','PdH','AgH')
    set(gca,'FontSize',24,'FontWeight','bold','linewidth',3)
    axis([0.0,6,-1,2.5])
    fig = figure(22);
    fig.Units = 'inches';
    fig.Position = [2 2 fig_width fig_height];

     % Comparison Two Body Potential
    figure(23)
    plot(r,Phi_PdPd,'k-',r,Phi_AgAg,'k:',r,Phi_PdAg,'k--','markersize',3,
'linewidth',4)
    xlabel('r (A^o)','fontsize',24)
    ylabel('Two Body Potential (eV)','fontsize',24)
    legend('PdPd','AgAg','PdAg')
    set(gca,'FontSize',24,'FontWeight','bold','linewidth',3)
    axis([1.5,6,-.3,1])
    fig = figure(23);
    fig.Units = 'inches';
    fig.Position = [2 2 fig_width fig_height];

    x = [1:12]';
```

```
     % Cohesive Energy Plot for all structures
    figure (3)

plot(x,Zhou_Experimental,'b*',x,Model,'kO',x,Zhou,'bs',x,Siesta,'blackd','mar
kersize',10, 'linewidth',3)
    set(gca,'FontSize',12,'FontWeight','bold','linewidth',3)
    xlabel('x','fontsize',24)
    ylabel('E_c (eV/atom)','fontsize',24)
    legend('Zhou DFT','Model','Zhou MD','Siesta','Location','Northwest')
    legend({},'fontsize',18)
    xticks([1 2 3 4 5 6 7 8 9 10 11 12])
%     axis([0,13,-2.8,-3.5])
    xticklabels({'PdAgH025 O1','PdAgH025 O2','PdAgH025 TE1','PdAgH050 O1
O2',...
        'PdAgH050 O2 O2','PdAgH050 TE1 TE2','PdAgH050 TE1 TE8','PdAgH075 O1
O2 O2',...
        'PdAgH075 O2 O2 O2','PdAgH075 T','PdAgH100 O1 O2 O2 O2','PdAgH100
T'})
    xtickangle(45)
    fig = figure(3);
    fig.Units = 'inches';
    fig.Position = [2 2 fig_width fig_height];

    x = [1:11]';

    Siesta = [Ec_exp_PdAgH025_O1; Ec_exp_PdAgH025_O2; Ec_exp_PdAgH025_T;...
           Ec_exp_PdAgH050_O1_O2; Ec_exp_PdAgH050_O2_O2;
Ec_exp_PdAgH050_T;...
           Ec_exp_PdAgH075_O1_O2_O2; Ec_exp_PdAgH075_O2_O2_O2;
Ec_exp_PdAgH075_T;...
           Ec_exp_PdAgH100_O1_O2_O2_O2; Ec_exp_PdAgH100_T];

    Fitting = [Ec_Pd075Ag025H025_O1; Ec_Pd075Ag025H025_O2; ...
            Ec_Pd075Ag025H050_O1_O2;Ec_Pd075Ag025H050_O2_O2;...
            Ec_Pd075Ag025H075_O1_O2_O2;Ec_Pd075Ag025H075_O2_O2_O2;...
            Ec_Pd075Ag025H100_O1_O2_O2_O2]';

    Model        = [Ec_Pd075Ag025H050_TE1_TE2;Ec_Pd075Ag025H050_TE1_TE8;...
        Ec_Pd075Ag025H050_TE3_TE4;Ec_Pd075Ag025H050_TE3_TE6;...
            Ec_Pd075Ag025H075_TE5_TE6_TE7;Ec_Pd075Ag025H075_TE1_TE2_TE3;...

Ec_Pd075Ag025H075_TE1_TE2_TE8;Ec_Pd075Ag025H100_TE1_TE5_TE6_TE7;...

Ec_Pd075Ag025H100_TE1_TE2_TE3_TE4;Ec_Pd075Ag025H100_TE1_TE2_TE7_TE8];

    Siesta = [Ec_exp_PdAgH025_O1; Ec_exp_PdAgH025_O2; Ec_exp_PdAgH025_T;...
           Ec_exp_PdAgH050_O1_O2; Ec_exp_PdAgH050_O2_O2; Ec_exp_PdAgH050_T;
Ec_exp_PdAgH050_T;...
           Ec_exp_PdAgH075_O1_O2_O2; Ec_exp_PdAgH075_O2_O2_O2;
Ec_exp_PdAgH075_T;...
           Ec_exp_PdAgH100_O1_O2_O2_O2; Ec_exp_PdAgH100_T];

    Zhou_Experimental = [-3.396;-3.337;-3.373;-3.180;-3.147;-3.171;-3.171;-
3.035;-2.944;-3.025;-2.863;-2.919];
```

```
    Zhou = [-3.49499221;-3.2892551;-3.4706926;-3.23099;-3.086482;-3.168897;-
3.168897;-3.070289;-2.9575078;-3.031653827;-2.95657414;-3.0171908];


    Rownames = {'PdAgH050_TE1_TE2','PdAgH050_TE1_TE8','PdAgH050_TE3_TE4',...
        'PdAgH050_TE3_TE8','PdAgH075_TE5_TE6_TE7','PdAgH075_TE1_TE2_TE3',...
        'PdAgH075_TE1_TE2_TE8','PdAgH100_TE1_TE5_TE6_TE7',...
        'PdAgH100_TE1_TE2_TE3_TE4','PdAgH100_TE1_TE2_TE7_TE8'};

    Columnnames = {'Model'};

    Results = table(Model,'VariableNames',Columnnames,'RowNames',Rownames)

end


%*************************************************************************
%
%*************************************************************************
function [Ec_,dEda_]=
fitProperties_1(r,a,N,index,t1,y,f_11,dfdr_11,d2fdr2_11,X0,...

F,dFdrho,d2Fdrho2,X1,phi_11,dphidr_11,d2phidr2_11,X2,rc_1,...

t2,x,f_22,dfdr_22,d2fdr2_22,X3,phi_12,dphidr_12,d2phidr2_12,X4,rc_2)

    Ec_ =
Ec(r,a,index,t1,y,f_11,X0,F,X1,phi_11,X2,rc_1,t2,x,f_22,X3,phi_12,X4,rc_2);

    dEda_ =
dEda(r,a,index,t1,y,f_11,dfdr_11,X0,dFdrho,X1,dphidr_11,X2,rc_1,...
                        t2,x,f_22,dfdr_22,X3,dphidr_12,X4,rc_2);
end

%*************************************************************************
% Property Functions Used In Parameters Fitting
%*************************************************************************
function [Ec_,dEda_,C11,C12,C44,Bm_,Ev_] = fitProperties(r,a,N,index,...
          t1,y,f_11,dfdr_11,d2fdr2_11,X0,F,dFdrho,d2Fdrho2,X1,phi_11,...
          dphidr_11,d2phidr2_11,X2,rc_1,t2,x,f_22,dfdr_22,d2fdr2_22,X3,...
          phi_12,dphidr_12,d2phidr2_12,X4,rc_2)
    % Cohesive Energy
    Ec_ = Ec(r,a,index,t1,y,f_11,X0,F,X1,phi_11,X2,rc_1,t2,x,f_22,X3,...
             phi_12,X4,rc_2);
    % First Derivative of Ec with respect to a
    dEda_ = dEda(r,a,index,t1,y,f_11,dfdr_11,X0,dFdrho,X1,dphidr_11,X2,...
                 rc_1,t2,x,f_22,dfdr_22,X3,dphidr_12,X4,rc_2);
    % C11 Elastic Constant
    C11 = Cijkl(1,1,1,1,r,a,N,index,t1,y,f_11,dfdr_11,d2fdr2_11,X0,...
                dFdrho,d2Fdrho2,X1,dphidr_11,d2phidr2_11,X2,rc_1,...

t2,x,f_22,dfdr_22,d2fdr2_22,X3,dphidr_12,d2phidr2_12,X4,rc_2);
    % C12 Elastic Constant
    C12 = Cijkl(1,1,2,2,r,a,N,index,t1,y,f_11,dfdr_11,d2fdr2_11,X0,...
```

```
                dFdrho,d2Fdrho2,X1,dphidr_11,d2phidr2_11,X2,rc_1,...

t2,x,f_22,dfdr_22,d2fdr2_22,X3,dphidr_12,d2phidr2_12,X4,rc_2);
    % C44 Elastic Constant
    C44 = Cijkl(2,3,2,3,r,a,N,index,t1,y,f_11,dfdr_11,d2fdr2_11,X0,...
                dFdrho,d2Fdrho2,X1,dphidr_11,d2phidr2_11,X2,rc_1,...

t2,x,f_22,dfdr_22,d2fdr2_22,X3,dphidr_12,d2phidr2_12,X4,rc_2);
    % Bulk Modulus
    Bm_ = Bm(r,a,N,index,t1,y,f_11,dfdr_11,d2fdr2_11,X0,...
             dFdrho,d2Fdrho2,X1,d2phidr2_11,X2,rc_1,t2,x,f_22,dfdr_22,...
             d2fdr2_22,X3,d2phidr2_12,X4,rc_2);
    % Vacancy Formation
    Ev_ = 0;
    % Ev_ = Ev(index,t1,a,phi_11,X2,f_11,X0,F,X1,r,rc_1);
end

%**************************************************************************
% Cohesive Energy for Interstitial Solid Solution
%**************************************************************************
function f = Ec(ri,a,index,t1,y,f_11,X0,F_11,X1,phi_11,X2,rc_1,t2,x,...
                f_22,X3,phi_12,X4,rc_2)
    %
    f = 0;
    Rho_12 = 0;
    Phi_12 = 0;
    %
    Rho_11 = y * Rho(index,t1,ri,a,rc_1,f_11,X0);
    Phi_11 = y * Phi(index,t1,ri,a,rc_1,phi_11,X2);
    %
    if nargin == 19
       Rho_12  = x * Rho(index,t2,ri,a,rc_2,f_22,X3);
       Phi_12  = x * Phi(index,t2,ri,a,rc_2,phi_12,X4);
    end
    %
    Rho_1 = Rho_11 + Rho_12;
    F_1 = F_11(Rho_1,X1);
    f = F_1 + 0.5 * Phi_11 + 0.5 * Phi_12;
end



function f = Ec3(ri,a,index,t1,y,f_11,X0,F_11,X1,phi_11,X2,rc_1,t2,x,...
                f_22,X3,phi_12,X4,rc_2,t3,z,f_33,X5,phi_13,X6,rc_3)
    %
    f = 0;
    Rho_12 = 0;
    Phi_12 = 0;
    Rho_13 = 0;
    Phi_13 = 0;
    %
    Rho_11 = y * Rho(index,t1,ri,a,rc_1,f_11,X0);
    Phi_11 = y * Phi(index,t1,ri,a,rc_1,phi_11,X2);
    %
    if nargin >= 19
       Rho_12  = x * Rho(index,t2,ri,a,rc_2,f_22,X3);
```

```
       Phi_12  = x * Phi(index,t2,ri,a,rc_2,phi_12,X4);
    end

    if nargin >= 26
       Rho_13  = z * Rho(index,t3,ri,a,rc_3,f_33,X5);
       Phi_13  = z * Phi(index,t3,ri,a,rc_3,phi_13,X6);
    end
    %
    Rho_1 = Rho_11 + Rho_12 + Rho_13;
    F_1 = F_11(Rho_1,X1);
    f = F_1 + 0.5 * Phi_11 + 0.5 * Phi_12 + 0.5 * Phi_13;
end


%************************************************************************
% Elestic Constants for an Interstitial Solid Solution
%************************************************************************
function f = Cijkl(i,j,k,l,r,a,N,index,t1,y,f_11,dfdr_11,d2fdr2_11,X0,...
                   dFdrho,d2Fdrho2,X1,dphidr_11,d2phidr2_11,X2,rc_1,t2,...
                   x,f_22,dfdr_22,d2fdr2_22,X3,dphidr_12,d2phidr2_12,X4,rc_2)
    %
    Rho_12 = 0;
    Vij_12 = 0;
    Vkl_12 = 0;
    Wijkl_12 = 0;
    Bijkl_12 = 0;
    %
    Rho_11   = y * Rho(index,t1,r,a,rc_1,f_11,X0);
    Vij_11   = y * Vij(index,t1,r,a,rc_1,dfdr_11,X0,i,j,0,0);
    Vkl_11   = y * Vij(index,t1,r,a,rc_1,dfdr_11,X0,0,0,k,l);
    Wijkl_11 = y * Wijkl(index,t1,r,a,rc_1,dfdr_11,d2fdr2_11,X0,i,j,k,l);
    Bijkl_11 = y * Bijkl(index,t1,r,a,rc_1,dphidr_11,d2phidr2_11,X2,i,j,k,l);
    %
    if nargin == 31
        Rho_12   = x * Rho(index,t2,r,a,rc_2,f_22,X3);
        Vij_12   = x * Vij(index,t2,r,a,rc_2,dfdr_22,X3,i,j,0,0);
        Vkl_12   = x * Vij(index,t2,r,a,rc_2,dfdr_22,X3,0,0,k,l);
        Wijkl_12 = x * Wijkl(index,t2,r,a,rc_2,dfdr_22,d2fdr2_22,X3,i,j,k,l);
        Bijkl_12 = x *
Bijkl(index,t2,r,a,rc_2,dphidr_12,d2phidr2_12,X4,i,j,k,l);
    end
    Rho_1      = Rho_11 + Rho_12;
    dFdrho_1   = dFdrho(Rho_1,X1);
    d2Fdrho2_1 = d2Fdrho2(Rho_1,X1);
    % Elastic constants Cijkl
    V = (a^3)/N;
    Cijkl = (1/V)*( d2Fdrho2_1*(Vij_11 + Vij_12)*(Vkl_11 + Vkl_12) +...
                    dFdrho_1*(Wijkl_11 + Wijkl_12) + Bijkl_11 + Bijkl_12 );
    f = Cijkl * 1.602176462;
end

%************************************************************************
% Stress for an Interstitial solid solution
%************************************************************************
function f =
S(i,j,r,a,N,index,t1,y,f_11,dfdr_11,X0,dFdrho,X1,dphidr_11,X2,rc_1,...
```

```
            t2,x,f_22,dfdr_22,X3,dphidr_12,X4,rc_2)

        Rho_12 = 0;
        Vij_12 = 0;
        Aij_12 = 0;
        %
        Rho_11 = y * Rho(index,t1,r,a,rc_1,f_11,X0);
        Vij_11 = y * Vij(index,t1,r,a,rc_1,dfdr_11,X0,i,j,0,0);
        Aij_11 = y * Aij(index,t1,r,a,rc_1,dphidr_11,X2,i,j,0,0);
        if nargin == 24
            Rho_12 = x * Rho(index,t2,r,a,rc_2,f_22,X3);
            Vij_12 = x * Vij(index,t2,r,a,rc_2,dfdr_22,X3,i,j,0,0);
            Aij_12 = x * Aij(index,t2,r,a,rc_2,dphidr_12,X4,i,j,0,0);
        end
        Rho_1     = Rho_11 + Rho_12;
        dFdrho_1  = dFdrho(Rho_1,X1);
        V = (a^3)/N;
        f = (1/V) * (Aij_11 + Aij_12 + dFdrho_1 * (Vij_11 + Vij_12));
end

%*************************************************************************
% First derivative of Cohesive Energy with respect to a for an
% Interstitial solid solution
%*************************************************************************
function  f =
dEda(ri,a,index,t1,y,f_11,dfdr_11,X0,dFdrho,X1,dphidr_11,X2,rc_1,...
                  t2,x,f_22,dfdr_22,X3,dphidr_12,X4,rc_2)
    %
    Rho_12    = 0;
    dRhoda_12 = 0;
    dPhida_12 = 0;
    %
    Rho_11    = y * Rho(index,t1,ri,a,rc_1,f_11,X0);
    dRhoda_11 = y * dRhoda(index,t1,ri,a,rc_1,dfdr_11,X0);
    dPhida_11 = y * dPhida(index,t1,ri,a,rc_1,dphidr_11,X2);
    %
    if nargin == 21
       Rho_12    = x * Rho(index,t2,ri,a,rc_2,f_22,X3);
       dRhoda_12 = x * dRhoda(index,t2,ri,a,rc_2,dfdr_22,X3);
       dPhida_12 = x * dPhida(index,t2,ri,a,rc_2,dphidr_12,X4);
    end
    %
    Rho_1     = Rho_11    + Rho_12;
    dRhoda_1  = dRhoda_11 + dRhoda_12;
    dFda_1 = dFda(dFdrho,Rho_1,dRhoda_1,X1);
    f = dFda_1 + 0.5 * dPhida_11 + 0.5 * dPhida_12;
end

%*************************************************************************
% Second derivative of Cohesive Energy with respect to a for Interstitial
% Solid Solution
%*************************************************************************
function  f = d2Eda2(ri,a,index,t1,y,f_11,dfdr_11,d2fdr2_11,X0,...
                     dFdrho,d2Fdrho2,X1,d2phidr2_11,X2,rc_1,t2,x,...
                     f_22,dfdr_22,d2fdr2_22,X3,d2phidr2_12,X4,rc_2)
    %
```

```
    Rho_12      = 0;
    dRhoda_12   = 0;
    d2Rhoda2_12 = 0;
    d2Phida2_12 = 0;
    %
    Rho_11      = y * Rho(index,t1,ri,a,rc_1,f_11,X0);
    dRhoda_11   = y * dRhoda(index,t1,ri,a,rc_1,dfdr_11,X0);
    d2Rhoda2_11 = y * d2Rhoda2(index,t1,ri,a,rc_1,d2fdr2_11,X0);
    d2Phida2_11 = y * d2Phida2(index,t1,ri,a,rc_1,d2phidr2_11,X2);
    %
    if nargin == 24
       Rho_12      = x * Rho(index,t2,ri,a,rc_2,f_22,X3);
       dRhoda_12   = x * dRhoda(index,t2,ri,a,rc_2,dfdr_22,X3);
       d2Rhoda2_12 = x * d2Rhoda2(index,t2,ri,a,rc_2,d2fdr2_22,X3);
       d2Phida2_12 = x * d2Phida2(index,t2,ri,a,rc_2,d2phidr2_12,X4);
    end
    Rho_1       = Rho_11 + Rho_12;
    dRhoda_1    = dRhoda_11 + dRhoda_12;
    d2Rhoda2_1  = d2Rhoda2_11 + d2Rhoda2_12;
    d2Fda2_1 = d2Fda2(dFdrho,d2Fdrho2,Rho_1,dRhoda_1,d2Rhoda2_1,X1);
    f = d2Fda2_1 + 0.5 * d2Phida2_11 + 0.5 * d2Phida2_12;
end

%**************************************************************************
% Bulk modulous
%**************************************************************************
function f = Bm(ri,a,N,index,t1,y,f_11,dfdr_11,d2fdr2_11,X0,...
                       dFdrho,d2Fdrho2,X1,d2phidr2_11,X2,rc_1,t2,x,f_22,...
                       dfdr_22,d2fdr2_22,X3,d2phidr2_12,X4,rc_2)
    d2Eda2_ = 0;
    %
    if nargin == 16
       d2Eda2_ = d2Eda2(ri,a,index,t1,y,f_11,dfdr_11,d2fdr2_11,X0,...
                           dFdrho,d2Fdrho2,X1,d2phidr2_11,X2,rc_1);
    end
    if nargin == 25
        d2Eda2_ = d2Eda2(ri,a,index,t1,y,f_11,dfdr_11,d2fdr2_11,X0,...
                       dFdrho,d2Fdrho2,X1,d2phidr2_11,X2,rc_1,t2,x,f_22,...
                       dfdr_22,d2fdr2_22,X3,d2phidr2_12,X4,rc_2);
    end
    V = a^3/N;
    f = ((a)^2/(9.0*V))*d2Eda2_;
    f = f * 1.602176462;
end

%**************************************************************************
% Vacancy formation Energy
%**************************************************************************
function f = Ev(index,type,a,phi,X0,f,X1,F,X2,rij,rc)
    sum0 = 0;
    sum1 = 0;
    sum2 = 0;
    rho = Rho(index,type,rij,a,rc,f,X1);
    n = length(rij(:,1));
    for i = 1:n
        id = rij(i,1);
```

```
        if i ~= index && id == type
            l1 = (a/2)*rij(i,2);
            l2 = (a/2)*rij(i,3);
            l3 = (a/2)*rij(i,4);
            ri = sqrt(l1^2+l2^2+l3^2);
            if ri <= rc
                q0 = phi(ri,X0);
                f1 = f(ri,X1);
                sum0 = sum0 + q0;
                sum1 = sum1 + F(rho,X2);
                sum2 = sum2 + F(rho-f1,X2);
            end
        end
    end
    f = -0.5 * sum0 + sum2 - sum1;
end

%*************************************************************************
% The Embedding Atom Model EAM
%*************************************************************************

% Total pair potential energy for a central atom as function of interatomic
% distance
% index: array index of central atom
% type : Id of atom type
% ri   : atoms position in Lattice Crystal Structure
% a    : the lattice constant
% rc   : the cuttof radius
% phi  : the pair potential function
% X    : array of pair potential parameters
function f = Phi(index,type,ri,a,rc,phi,X)
    f = Sumfunc(index,type,ri,a,rc,phi,X);
end

% Total first derivative of pair potential with respect to interatomic
% distance rij
function f = dPhidr(index,type,ri,a,rc,dphidr,X)
    f = Sumfunc(index,type,ri,a,rc,dphidr,X);
end

% Total second derivative of pair potential with respect to interatomic
% distance rij
function f = d2Phidr2(index,type,ri,a,rc,d2phidr2,X)
    f = Sumfunc(index,type,ri,a,rc,d2phidr2,X);
end

% Total first derivative of pair potential with respect to a
function f = dPhida(index,type,ri,a,rc,dphidr,X)
    f = SumfuncM(index,type,ri,a,rc,dphidr,X);
end

% Total second derivative of pair potential with respect to a
function f = d2Phida2(index,type,ri,a,rc,d2phidr2,X)
    f = SumfuncM2(index,type,ri,a,rc,d2phidr2,X);
end
```

```
% Total Electron Density for a central atom as function of interatomic
% distance
% index: array index of central atom
% type : Id of atom type
% ri   : atoms position in Lattice Crystal Structure
% a    : the lattice constant
% rc   : the cuttof radius
% f    : the atomic density function
% X    : array of atomic density function parameters
function f = Rho(index,type,ri,a,rc,f,X)
    f = Sumfunc(index,type,ri,a,rc,f,X);
end

% Total first derivative of electron density with respect to interatomic
% distance rij
function f = dRhodr(index,type,ri,a,rc,dfdr,X)
    f = Sumfunc(index,type,ri,a,rc,dfdr,X);
end

% Total second derivative of electron density with respect to interatomic
% distance rij
function f = d2Rhodr2(index,type,ri,a,rc,d2fdr2,X)
    f = Sumfunc(index,type,ri,a,d2fdr2,X);
end

% Total first derivative of electron density with respect a
function f = dRhoda(index,type,ri,a,rc,dfdr,X)
    f = SumfuncM(index,type,ri,a,rc,dfdr,X);
end

% Total second derivative of electron density with respect a
function f = d2Rhoda2(index,type,ri,a,rc,d2fdr2,X)
    f = SumfuncM2(index,type,ri,a,rc,d2fdr2,X);
end

% Total first derivative of electron density with respect ri,rj,rk, and rl
function f = dRhodrij(index,type,ri,a,rc,dfdr,X,i,j,k,l)
     f = Sumfuncrijkl(index,type,ri,a,rc,dfdr,X,i,j,k,l,1);
end

% second derivative of electron density of rij
function f = d2Rhodr2ijkl(index,type,ri,a,rc,dfdr,X,i,j,k,l)
     f = Sumfuncrijkl(index,type,ri,a,rc,dfdr,X,i,j,k,l,2);
end

% First derivative of Atomic Electron Density with respect to ri
function f = dfdri(index,type,ri,a,rc,dfdr,X,i,j,k,l)
     f = Sumfuncrijkl(index,type,ri,a,rc,dfdr,X,i,j,k,l,1);
end

%  First derivative of Atomic Electron Density with respect to ri,rj,rk
%  and rl
function f = dfdrijkl(index,type,ri,a,rc,dfdr,X,i,j,k,l)
     f = Sumfuncrijkl(index,type,ri,a,rc,dfdr,X,i,j,k,l,3);
```

```
end

%  Second derivative of Atomic Electron Density with respect to ri,rj,rk
%  and rl
function f = d2fdr2ijkl(index,type,ri,a,rc,d2fdr2,X,i,j,k,l)
     f = Sumfuncrijkl(index,type,ri,a,rc,d2fdr2,X,i,j,k,l,2);
end

% First derivative of pair potential with respect to ri
function f = dphidri(index,type,ri,a,rc,dphidr,X,i,j,k,l)
     f = Sumfuncrijkl(index,type,ri,a,rc,dphidr,X,i,j,k,l,1);
end

% First derivative of pair potential with respect to ri,rj,rk and rl
function f = dPhidrijkl(index,type,ri,a,rc,dphidr,X,i,j,k,l)
     f = Sumfuncrijkl(index,type,ri,a,rc,dphidr,X,i,j,k,l,3);
end

% Second derivative of pair potential with respect to ri,rj,rk and rl
function f = d2Phidr2ijkl(index,type,ri,a,rc,d2phidr2,X,i,j,k,l)
     f = Sumfuncrijkl(index,type,ri,a,rc,d2phidr2,X,i,j,k,l,2);
end

% Function Used in the calculations of Elastic Constants at equlibrum
function f = Vij(index,type,ri,a,rc,dfdr,X,i,j,k,l)
    f = delta(i,j) * delta(k,l)*dfdri(index,type,ri,a,rc,dfdr,X,i,j,k,l);
end

% Function Used in the calculations of Elastic Constants at equlibrum
function f = Wijkl(index,type,ri,a,rc,dfdr,d2fdr2,X,i,j,k,l)
    f = d2fdr2ijkl(index,type,ri,a,rc,d2fdr2,X,i,j,k,l) - ...
        dfdrijkl(index,type,ri,a,rc,dfdr,X,i,j,k,l)+ ...
        delta(i,l) * dfdri(index,type,ri,a,rc,dfdr,X,0,0,k,l);
end

% Function Used in the calculations of Elastic Constants at equlibrum
function f = Bijkl(index,type,ri,a,rc,dphidr,d2phidr2,X,i,j,k,l)
     f = d2Phidr2ijkl(index,type,ri,a,rc,d2phidr2,X,i,j,k,l) - ...
           dPhidrijkl(index,type,ri,a,rc,dphidr,X,i,j,k,l)+ ...
         delta(i,l) * dphidri(index,type,ri,a,rc,dphidr,X,0,0,k,l);
     f = 0.5 * f;
end

%*************************************************************************
%   Park Hijazi Pd EAM Potential and Derivatives
%*************************************************************************

% Pd Embedding Energy Function as a function of electron density
function f = F_Pd(rho,X)
    f    = 0;
    Fe   = X(1);
    rhoe = X(2);
    n    = X(3);
    if rho > 0
        f = -Fe*(1-n*log(rho/rhoe))*(rho/rhoe)^n;
```

```
    end
end

% First derivative of Pd Embedding Energy Function with respect to
% electron density
function f = dFdrho_Pd(rho,X)
    f    = 0;
    Fe   = X(1);
    rhoe = X(2);
    n    = X(3);
    if rho > 0
        F = F_Pd(rho,X);
        f = (Fe*n/rho)*(rho/rhoe)^n+(n/rhoe)*F*(rho/rhoe)^(-1);
    end
end

% Second derivative of Pd Embedding Energy Function with respect to
% electron density
function f = d2Fdrho2_Pd(rho,X)
    f    = 0;
    Fe   = X(1);
    rhoe = X(2);
    n    = X(3);
    if rho > 0
        F  = F_Pd(rho,X);
        dF = dFdrho_Pd(rho,X);
        f  = -(n/rho^2)*Fe*(rho/rhoe)^n+(n^2/rho)*(1/rhoe)*Fe*(rho/rhoe)^(n-
1)+...
             (n/rhoe)*dF*(rho/rhoe)^(-1)-(n/rhoe^2)*F*(rho/rhoe)^(-2);
    end
end

% First derivative of Pd Embedding Energy Function with respect to rij
function f = dFdr(dFdrho,rho,dRhodr,X)
    f = dFdrho(rho,X) * dRhodr;
end

% Second derivative of Pd Embedding Energy Function with respect to rij
function f = d2Fdr2(dFdrho,d2Fdrho2,rho,dRhodr,d2Rhodr2,X)
    f = d2Fdrho2(rho,X) * dRhodr^2  + dFdrho(rho,X) * d2Rhodr2;
end

% First derivative of Pd Embedding Energy Function with respect to a
function f = dFda(dFdrho,rho,dRhoda,X)
    f = dFdrho(rho,X) * dRhoda;
end

% Second derivative of Pd Embedding Energy Function with respect to a
function f = d2Fda2(dFdrho,d2Fdrho2,rho,dRhoda,d2Rhoda2,X)
    f = d2Fdrho2(rho,X) * dRhoda^2  + dFdrho(rho,X) * d2Rhoda2;
end

% Pd Atomic Electron Density as a function of interatomic distance rij
function f = f_PdPd(rij,X)
    fe = X(1);
```

```
    Xi = X(2);
    re = X(3);
    f = fe*exp(-Xi*(rij-re));
end

%  First derivative Pd Atomic Electron Density function with respect to rij
function f = dfdr_PdPd(rij,X)
    fe = X(1);
    Xi = X(2);
    re = X(3);
    f = -fe*Xi*exp(-Xi*(rij-re));
end

%  First derivative Pd Atomic Electron Density function with respect to rij
function f = d2fdr2_PdPd(rij,X)
    fe = X(1);
    Xi = X(2);
    re = X(3);
    f  = fe*Xi^2*exp(-Xi*(rij-re));
end

% Pd-Pd pair potential as a function of interatomic distance rij
function f = phi_PdPd(rij,X)
    phie = X(1);
    S    = X(2);
    B    = X(3);
    re   = X(4);
    f    = -phie*(1+S*(rij/re-1))*exp(-B*(rij/re-1));
end

% First derivative of Pd-Pd pair potential with respect to rij
function f = dphidr_PdPd(r,X)
    phie = X(1);
    S    = X(2);
    B    = X(3);
    re   = X(4);
    f    = (-1/re)*(phie*S*exp(-B*(r/re-1))+B*phi_PdPd(r,X));
end

% Second derivative of Pd-Pd pair potential with respect to rij
function f = d2phidr2_PdPd(r,X)
    phie = X(1);
    S    = X(2);
    B    = X(3);
    re   = X(4);
    f    = (B/re)*((phie*S/re)*exp(-B*(r/re-1))-dphidr_PdPd(r,X));
end

%*************************************************************************
% Hydrogen EAM Potential and Derivatives
%*************************************************************************

% H-H unnormalized pair potential as a function of interatomic distance rij
function f = phi_HH_u(rij,X)
    DHH     = X(1);
```

```
    alphaHH = X(2);
    betaHH  = X(3);
    r0HH    = X(4);
    f = DHH*(betaHH*exp(-alphaHH*(rij-r0HH)) - ...
        alphaHH*exp(-betaHH*(rij-r0HH)));
end

% First derivative of H-H unnormalized pair potential with respect to rij
function f = dphidr_HH_u(r,X)
    DHH     = X(1);
    alphaHH = X(2);
    betaHH  = X(3);
    r0HH    = X(4);
    f = DHH*( -alphaHH * betaHH*exp(-alphaHH*(r-r0HH)) +...
        betaHH*alphaHH*exp(-betaHH*(r-r0HH)) );
end

% Secon derivative of H-H unnormalized pair potential with respect to rij
function f = d2phidr2_HH_u(r,X)
     DHH     = X(1);
     alphaHH = X(2);
     betaHH  = X(3);
     r0HH    = X(4);
     f = DHH*( (alphaHH^2) * betaHH*exp(-alphaHH*(r-r0HH)) - ...
         (betaHH^2)*alphaHH*exp(-betaHH*(r-r0HH)) );
end

% H-H normalized pair potential as a function of interatomic distance rij
function f = phi_HH(rij,X)
    X2_HH = X(1:1:4);
    X0_HH = X(5:1:6);
    X1_HH = X(7:1:11);
    rho0H = X(11);
    f = phi_HH_u(rij,X2_HH) + 2 * dFdrho_H_u(rho0H,X1_HH) * f_HH(rij,X0_HH);
end

% First derivative of H-H normalized pair potential with respect to rij
function f = dphidr_HH(rij,X)
    X2_HH = X(1:1:4);
    X0_HH = X(5:1:6);
    X1_HH = X(7:1:11);
    rho0H = X(11);
    f = dphidr_HH_u(rij,X2_HH) + 2 * dFdrho_H_u(rho0H,X1_HH) * ...
        dfdr_HH(rij,X0_HH);
end

% Second derivative of H-H normalized pair potential with respect to rij
function f = d2phidr2_HH(rij,X)
    X2_HH = X(1:1:4);
    X0_HH = X(5:1:6);
    X1_HH = X(7:1:11);
    rho0H = X(11);
    f = d2phidr2_HH_u(rij,X2_HH) + 2 * dFdrho_H_u(rho0H,X1_HH) * ...
        d2fdr2_HH(rij,X0_HH);
end
```

```
% H Atomic Electron Density as a function of interatomic distance
function f = f_HH(r,X)
    CH = X(1);
    DH = X(2);
    f = CH*exp(-DH*r);
end

% First derivative H Atomic Electron Density with respect to rij
function f = dfdr_HH(r,X)
    CH = X(1);
    DH = X(2);
    f = -DH*CH*exp(-DH*r);
end

% Second derivative H Atomic Electron Density with respect to rij
function f = d2fdr2_HH(r,X)
    CH = X(1);
    DH = X(2);
    f = (DH^2)*CH*exp(-DH*r);
end

% H unormalized Embedding Energy function
function f = F_H_u(rho,X)
    aH = X(1);
    bH = X(2);
    cH = X(3);
    dH = X(4);
    EH = 0.0540638;
%     EH = 0.055;
    f = -cH * ( (1/(2+dH))*(rho+EH)^(2+dH) - ((aH + bH)/(1+dH))*...
        (rho+EH)^(1+dH) + ((aH * bH)/dH) *(rho+EH)^dH );
end

% First derivative of H unnormalized Embedding function with respect to Rho
function f = dFdrho_H_u(rho,X)
    aH = X(1);
    bH = X(2);
    cH = X(3);
    dH = X(4);
    EH = 0.0540638;
%     EH = 0.055;
    f = -cH * ( (rho+EH)^(1+dH) - (aH + bH) * (rho+EH)^(dH) +...
        (aH * bH) *(rho+EH)^(dH-1) );
end

% Second derivative of H unnormalized Embedding function with respect to Rho
function f = d2Fdrho2_H_u(rho,X)
    aH = X(1);
    bH = X(2);
    cH = X(3);
    dH = X(4);
    EH = 0.0540638;
%     EH = 0.055;
    f = -cH * ( (1+dH)*(rho+EH)^(dH) - dH*(aH + bH) * (rho+EH)^(dH-1) + ...
        (dH-1)*(aH * bH) *(rho+EH)^(dH-2) );
end
```

```
% H normalized Embedding Energy function
function f = F_H(rho,X)
    X1_HH = X(7:1:10);
    rho0H = X(11);
    f = F_H_u(rho,X1_HH) - dFdrho_H_u(rho0H,X1_HH) * rho;
end

% First derivative of H normalized Embedding function with respect to Rho
function f = dFdrho_H(rho,X)
    X1_HH = X(7:1:10);
    rho0H = X(11);
    f = dFdrho_H_u(rho,X1_HH) - dFdrho_H_u(rho0H,X1_HH);
end

% Second derivative of H normalized Embedding function with respect to Rho
function f = d2Fdrho2_H(rho,X)
   X1_HH = X(7:1:10);
   f = d2Fdrho2_H_u(rho,X1_HH);
end

% Pd-H pair potential as a function of interatomic destance rij
function f = phi_PdH(r,X)
        DPdH      = X(1);
        alphaPdH  = X(2);
        betaPdH   = X(3);
        r0PdH     = X(4);
        f = DPdH*( betaPdH*exp(-alphaPdH*(r-r0PdH)) - ...
            alphaPdH*exp(-betaPdH*(r-r0PdH)) );
end

% First derivative of Pd-H pair potential with respect to interatomic
% destance rij
function f = dphidr_PdH(r,X)
        DPdH      = X(1);
        alphaPdH  = X(2);
        betaPdH   = X(3);
        r0PdH     = X(4);
        f = DPdH*( -alphaPdH*betaPdH*exp(-alphaPdH*(r-r0PdH)) +...
            betaPdH*alphaPdH*exp(-betaPdH*(r-r0PdH)) );
end

% Second derivative of Pd-H pair potential with respect to interatomic
% destance rij
function f = d2phidr2_PdH(r,X)
        DPdH      = X(1);
        alphaPdH  = X(2);
        betaPdH   = X(3);
        r0PdH     = X(4);
        f = DPdH*( (alphaPdH^2)*betaPdH*exp(-alphaPdH*(r-r0PdH)) - ...
            (betaPdH^2)*alphaPdH*exp(-betaPdH*(r-r0PdH)) );
end

%**************************************************************************
% Summation function
```

```
%*************************************************************************
function f = Sumfunc(index,type,ri,a,rc,func,X)
    f = 0;
    cl1 = (a/2)*ri(index,2);
    cl2 = (a/2)*ri(index,3);
    cl3 = (a/2)*ri(index,4);
    n = length(ri(:,1));
    for i = 1:n
        id = ri(i,1);
        if i ~= index && id == type
            l1 = (a/2)*ri(i,2);
            l2 = (a/2)*ri(i,3);
            l3 = (a/2)*ri(i,4);
            rij = sqrt((l1-cl1)^2 + (l2-cl2)^2 + (l3-cl3)^2);
            if rij <= rc
                f = f + func(rij,X);
            end
        end
    end
end

function f = SumfuncMP(index,type,ri,a,rc,func,X,p)
    f = 0;
    cl1 = (a/2)*ri(index,2);
    cl2 = (a/2)*ri(index,3);
    cl3 = (a/2)*ri(index,4);
    n = length(ri(:,1));
    for i = 1:n
        id = ri(i,1);
        if i ~= index && id == type
            l1 = (a/2)*ri(i,2);
            l2 = (a/2)*ri(i,3);
            l3 = (a/2)*ri(i,4);
            rij = sqrt((l1-cl1)^2 + (l2-cl2)^2 + (l3-cl3)^2);
            if rij <= rc
                f = f + func(rij,X) * (rij/a)^p;
            end
        end
    end
end

function f = SumfuncM(index,type,ri,a,rc,func,X)
    f = 0;
    cl1 = (a/2)*ri(index,2);
    cl2 = (a/2)*ri(index,3);
    cl3 = (a/2)*ri(index,4);
    n = length(ri(:,1));
    for i = 1:n
        id = ri(i,1);
        if i ~= index && id == type
            l1 = (a/2)*ri(i,2);
            l2 = (a/2)*ri(i,3);
            l3 = (a/2)*ri(i,4);
            rij = sqrt((l1-cl1)^2 + (l2-cl2)^2 + (l3-cl3)^2);
            if rij <= rc
                f = f + func(rij,X) * rij/a;
```

```
            end
        end
    end
end

%
function f = SumfuncM2(index,type,ri,a,rc,func,X)
    f = 0;
    cl1 = (a/2)*ri(index,2);
    cl2 = (a/2)*ri(index,3);
    cl3 = (a/2)*ri(index,4);
    n = length(ri(:,1));
    for i = 1:n
        id = ri(i,1);
        if i ~= index && id == type
            l1 = (a/2)*ri(i,2);
            l2 = (a/2)*ri(i,3);
            l3 = (a/2)*ri(i,4);
            rij = sqrt((l1-cl1)^2 + (l2-cl2)^2 + (l3-cl3)^2);
            if rij <= rc
                  f = f + func(rij,X) * (rij/a)^2;
            end
        end
    end
end

%
function f = Sumfuncrijkl(index,type,r,a,rc,func,X,i,j,k,l,p)
    f = 0;
    ri = 1;
    rj = 1;
    rk = 1;
    rl = 1;
    cl(1) = (a/2)*r(index,2);
    cl(2) = (a/2)*r(index,3);
    cl(3) = (a/2)*r(index,4);
    n = length(r(:,1));
    for ii = 1:n
        id = r(ii,1);
        if ii ~= index && id == type
            l1(1) = (a/2)*r(ii,2);
            l1(2) = (a/2)*r(ii,3);
            l1(3) = (a/2)*r(ii,4);
            rij = sqrt((l1(1)-cl(1))^2 + (l1(2)-cl(2))^2 + (l1(3)-cl(3))^2);
            if rij < rc
                if i ~= 0
                    ri = l1(i)-cl(i);
                end
                if j ~= 0
                    rj = l1(j)-cl(j);
                end
                 if k ~= 0
                    rk = l1(k)-cl(k);
                end
                if l ~= 0
                    rl = l1(l)-cl(l);
```

```
                end
                f = f + func(rij,X) * ri*rj*rk*rl/rij^p;
            end
        end
    end
end

%
function f = delta(i,j)
   if(i == j)
        f = 1;
     else
        f = 0;
     end
end


%************************************************************************
% Expermental data for FCC metals
%************************************************************************
function [afcce,Ecfcce,c11e,c12e,c44e,Bme,Eve,abcce,Ecbcce,cpre] =
parameters(el)
    index = 1;
    elements = ['Ag';'Al';'Au';'Cu';'Ni';'Pd';'Pt';'Hi'];
    for i = 1:length(elements)
        x = elements(i,:);
        if el == x
          index = i;
        end
    end
    afcc =  [4.09;4.05;4.08;3.615;3.52;3.89;3.92;3.38];
    Ecfcc = [2.85;3.36;3.93;3.54;4.45;3.91;5.77;2.119];
% % %  Elastic Constants in dyne/cm^2 10^12
    c11 =   [1.24;1.14;1.86;1.70;2.465;2.3412;3.47;0.0];
    c12 =   [0.934;0.619;1.57;1.225;1.473;1.7614;2.51;0.0];
    c44 =   [0.461;0.316;0.42;0.758;1.247;0.7117;0.765;0.05*1.6021766208];
    cpr =   [1;1;1;1;1;1;1;0.05*1.6021766208];
    % Unrelaxed
    Evfcc = [1.1;0.866;0.9;1.3;1.70;1.54;1.60;0.0];
    B =     [1.04;0.793;1.67;1.38;1.804;1.9547;2.83;0.5*1.6021766208];  %

    abcc =  [3.32;3.3;3.18;2.87;2.89;3.16;3.21;0.0];
    Ecbcc = [1;1;1;3.49;1;1;1;1];

    %
    afcce = afcc(index);
    Ecfcce = Ecfcc(index);
    Eve = Evfcc(index);
    c11e = c11(index);
    c12e = c12(index);
    c44e = c44(index);
    Bme = B(index);
    abcce = abcc(index);
    Ecbcce = Ecbcc(index);
    cpre   = cpr(index);
end
```

```
%************************************************************************
% Expermental and bb-initio data for PdH
%************************************************************************
function [afcce,Ecfcce,C44fcce,Cprfcce,Bmfcce] = PdH_parameters(el)
    index1 = 1;
    elements = ['Pd000H';'PdH000';'PdH025';'PdH050';'PdH075';...
                'PdH100';'PdHT50';'PdHT75'];
    [n,m]  = size(elements);
    for i = 1:n
        x = elements(i,:);
        if el == x
          index1 = i;
        end
    end
    a  =  [3.38;3.89;3.9421;4.0007;4.0593;4.1179;4.0007;4.0593];
    Ec =  [2.119;3.91;3.64024;3.48770;3.37031;3.26966;3.50000;3.50000];
% % %  Elastic Constants with conv from eV/A^3 to dyne/cm^2 10^12
    C44 = [0.05,0.7117/1.6021766208,0.459,0.42,0.382,0.3440,...
           0.3810,0.3810] * 1.6021766208;  % Pd from Rayne
    Cpr = [0.05,.2899/1.6021766208,0.055,0.135,0.164,0.146,...
           0.1630,0.1630] * 1.6021766208;  % Pd from Rayne
    Bm =  [0.50,1.9547/1.6021766208,1.05,1.05,1.05,1.05,...
           1.05,1.05] * 1.6021766208;  % Pd from Rayne, Zhou values added
    afcce  = a(index1);
    Ecfcce = Ec(index1);
    C44fcce  = C44(index1);
    Cprfcce  = Cpr(index1);
    Bmfcce  = Bm(index1);
end

 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 %   Pd Cutoff Transformed Functions
 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


function f = phi_PdPd_cut(rij,X)
    global rc_1;
    global NP;
    f = phi_PdPd(rij,X) - phi_PdPd(rc_1,X) + (rc_1/NP) * (1-(rij/rc_1)^NP )*
dphidr_PdPd(rc_1,X);
 end

function f = dphidr_PdPd_cut(rij,X)
    global rc_1;
    global NP;
    f = dphidr_PdPd(rij,X) - (rij/rc_1)^(NP-1) * dphidr_PdPd(rc_1,X);
end

function f = d2phidr2_PdPd_cut(rij,X)
    global rc_1;
    global NP;
    f = d2phidr2_PdPd(rij,X)- ((NP-1)/rc_1) * (rij/rc_1)^(NP-2) *
dphidr_PdPd(rc_1,X);
```

```
end

function f = f_PdPd_cut(rij,X)
    global rc_1;
    global NP;
    f = f_PdPd(rij,X) - f_PdPd(rc_1,X) + (rc_1/NP) * (1-(rij/rc_1)^NP) *
dfdr_PdPd(rc_1,X);
end

function f = dfdr_PdPd_cut(rij,X)
    global rc_1;
    global NP;
    f = dfdr_PdPd(rij,X) - (rij/rc_1)^(NP-1) * dfdr_PdPd(rc_1,X);
end

function f = d2fdr2_PdPd_cut(rij,X)
    global rc_1;
    global NP;
    f = d2fdr2_PdPd(rij,X)- ((NP-1)/rc_1) * (rij/rc_1)^(NP-2) *
dfdr_PdPd(rc_1,X);
end


 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 %   Hydrogen Cutoff Transformed functions
 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


function f = phi_HH_cut(rij,X)
    global rc_2;
    global NP2;
    f = phi_HH(rij,X) - phi_HH(rc_2,X) + (rc_2/NP2) * (1-(rij/rc_2)^NP2 )*
dphidr_HH(rc_2,X);
 end

function f = dphidr_HH_cut(rij,X)
    global rc_2;
    global NP2;
    f = dphidr_HH(rij,X) - (rij/rc_2)^(NP2-1) * dphidr_HH(rc_2,X);
end

function f = d2phidr2_HH_cut(rij,X)
    global rc_2;
    global NP2;
    f = d2phidr2_HH(rij,X)- ((NP2-1)/rc_2) * (rij/rc_2)^(NP2-2) *
dphidr_HH(rc_2,X);
end

function f = f_HH_cut(rij,X)
    global rc_2;
    global NP2;
    f = f_HH(rij,X) - f_HH(rc_2,X) + (rc_2/NP2) * (1-(rij/rc_2)^NP2) *
dfdr_HH(rc_2,X);
end
```

```
function f = dfdr_HH_cut(rij,X)
    global rc_2;
    global NP2;
    f = dfdr_HH(rij,X) - (rij/rc_2)^(NP2-1) * dfdr_HH(rc_2,X);
end

function f = d2fdr2_HH_cut(rij,X)
    global rc_2;
    global NP2;
    f = d2fdr2_HH(rij,X)- ((NP2-1)/rc_2) * (rij/rc_2)^(NP2-2) *
dfdr_HH(rc_2,X);
end

 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 %   PdH Cutoff Transformed Pair Potential
 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


function f = phi_PdH_cut(rij,X)
    global rc_3;
    global NP3;
    f = phi_PdH(rij,X) - phi_PdH(rc_3,X) + (rc_3/NP3) * (1-(rij/rc_3)^NP3 )*
dphidr_PdH(rc_3,X);
 end

function f = dphidr_PdH_cut(rij,X)
    global rc_3;
    global NP3;
    f = dphidr_PdH(rij,X) - (rij/rc_3)^(NP3-1) * dphidr_PdH(rc_3,X);
end

function f = d2phidr2_PdH_cut(rij,X)
    global rc_3;
    global NP3;
    f = d2phidr2_PdH(rij,X)- ((NP3-1)/rc_3) * (rij/rc_3)^(NP3-2) *
dphidr_PdH(rc_3,X);
end

 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 %   PdAg Pair Potential
 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

 function f = phi_PdAg(rij,X)
     X0_a = X(1:3);
     X0_b = X(4:6);
     X2_a = X(7:10);
     X2_b = X(11:14);
     f_a = f_PdPd(rij,X0_a);
     f_b = f_PdPd(rij,X0_b);
     phi_a = phi_PdPd(rij,X2_a);
     phi_b = phi_PdPd(rij,X2_b);
     f = .5*( f_b*f_a^-1*phi_a + f_a*f_b^-1*phi_b );
 end

 function f = dphidr_PdAg(rij,X)
```

```
     X0_a = X(1:3);
     X0_b = X(4:6);
     X2_a = X(7:10);
     X2_b = X(11:14);
     f_a = f_PdPd(rij,X0_a);
     f_b = f_PdPd(rij,X0_b);
     phi_a = phi_PdPd(rij,X2_a);
     phi_b = phi_PdPd(rij,X2_b);
     dfdr_a = dfdr_PdPd(rij,X0_a);
     dfdr_b = dfdr_PdPd(rij,X0_b);
     dphidr_a = dphidr_PdPd(rij,X2_a);
     dphidr_b = dphidr_PdPd(rij,X2_b);
     f = .5*( ((-1*f_b*dfdr_a*f_a^-2 + dfdr_b*f_a^-1)*phi_a + (f_b*f_a^-
1)*dphidr_a)...
           + ((-1*f_a*dfdr_b*f_b^-2 + dfdr_a*f_b^-1)*phi_b + (f_a*f_b^-
1)*dphidr_b) );
 end

 function f = d2phidr2_PdAg(rij,X)
     X0_a = X(1:3);
     X0_b = X(4:6);
     X2_a = X(7:10);
     X2_b = X(11:14);
     f_a = f_PdPd(rij,X0_a);
     f_b = f_PdPd(rij,X0_b);
     phi_a = phi_PdPd(rij,X2_a);
     phi_b = phi_PdPd(rij,X2_b);
     dfdr_a = dfdr_PdPd(rij,X0_a);
     dfdr_b = dfdr_PdPd(rij,X0_b);
     dphidr_a = dphidr_PdPd(rij,X2_a);
     dphidr_b = dphidr_PdPd(rij,X2_b);
     d2fdr2_a = d2fdr2_PdPd(rij,X0_a);
     d2fdr2_b = d2fdr2_PdPd(rij,X0_b);
     d2phidr2_a = d2phidr2_PdPd(rij,X2_a);
     d2phidr2_b = d2phidr2_PdPd(rij,X2_b);
     Phi_ab =
phi_ab(f_a,dfdr_a,d2fdr2_a,f_b,dfdr_b,d2fdr2_b,phi_a,dphidr_a,d2phidr2_a,phi_
b,dphidr_b,d2phidr2_b);
     Phi_ba =
phi_ab(f_b,dfdr_b,d2fdr2_b,f_a,dfdr_a,d2fdr2_a,phi_b,dphidr_b,d2phidr2_b,phi_
a,dphidr_a,d2phidr2_a);
     f = .5*(Phi_ab + Phi_ba);
 end

 function f =
phi_ab(f_a,dfdr_a,d2fdr2_a,f_b,dfdr_b,d2fdr2_b,phi_a,dphidr_a,d2phidr2_a,phi_
b,dphidr_b,d2phidr2_b)
    aa = -1*(dfdr_b*dfdr_a*f_a^-2 + f_b*d2fdr2_a*f_a^-2 -
2*f_b*(dfdr_a^2)*f_a^-3)...
     + (d2fdr2_b*f_a^-1 - dfdr_b*f_a^-2*dfdr_a);
    a = aa*phi_a + (-1*f_b*dfdr_a*f_a^-2 + dfdr_b*f_a^-1)*dphidr_a;
    b = dfdr_b*f_a^-1*dphidr_a - f_b*f_a^-2*dfdr_a*dphidr_a + f_b*f_a^-
1*d2phidr2_a;
    f = a + b;
 end
```

MATLAB PROGRAM FOR GENERATING DYNAMO FORMAT POTENTIAL FILE

```
function EAM_Potential_DYNAMO

    clc
    format short
    pwd

    global fId;


% % %   Pd and PdH
    x_Pd = [2.054085    0.216817    8.414105    7.221224    0.999999
3.316887];
    x_PdH = [0.589510   1.104827    0.942490    2.145808    0.942201...
    0.740938    2.373944    1.702142    8.370790    62.343273...
    0.000100    1.187000    1.300000    3.474173    4.903820];


    fId = fopen('PdAgH.eam.alloy','w');


% % %     re fit scaling only mixing rule potential
    S_Ag = 1.8319;
    S_Pd = 1.1063;

    % Pd Expermintal data
    [aFcce,EcFcce,c11e,c12e,c44e,~,Eve,aBcce,EcBcce] = parameters('Pd');

    % PdPd
    re = aFcce/sqrt(2);
    Fe = EcFcce-Eve;
    N  = 4;
    V  = aFcce^3/N;
    fe = EcFcce/V;
    x = [x_Pd, x_PdH];

    Xi   = x(1);
    phie = x(2);
    S    = x(3);
    B    = x(4);
    n    = x(5);
    rhoe = x(6);

    fe_S = S_Pd*EcFcce/V;
    rhoe_S = S_Pd*rhoe;

    % Phi_HH
    DHH =  x(7);
    aHH =  x(8);
    bHH =  x(9);
    % f_HH
    CH = x(10);
```

```
DH = x(11);

% Phi_PdH
DPdH = x(12);
aPdH = x(13);
bPdH = x(14);

% F_H
aH = x(15);
bH = x(16);
cH = x(17);
dH = x(18);
%
r0PdH = x(19);
r0HH  = x(20);
rho0H = x(21);

% Pd data
%  Rho PdPd fitting parameters
X0_PdPd = [fe_S,Xi,re];
%  F PdPd fitting parameters
X1_PdPd = [Fe,rhoe_S,n];
%  Phi PdPd fitting parametters
X2_PdPd = [phie,S,B,re];


%   F PdPd fitting parameters
X_F_PdPd   = [Fe,rhoe,n,0,0];
%   f Pd
X_f_PdPd   = [fe,Xi,re];
%   Phi PdPd fitting parametters
X_Phi_PdPd = [phie,S,B,re];
%

x_Ag = [1.584768032776473   0.154164182657381   8.491335427574112...
7.183185289310671   1.022270092304303   2.213230197455566];


% Ag Experimental data
[aFcce_Ag,EcFcce_Ag,C11e_Ag,C12e_Ag,C44e_Ag,Bme_Ag,Eve_Ag,aBcce_Ag,...
    EcBcce_Ag] = parameters('Ag');

re = aFcce_Ag/sqrt(2);
Fe = EcFcce_Ag-Eve_Ag;
N  = 4;
V  = aFcce_Ag^3/N;
fe = EcFcce_Ag/V;

% Ag Fitting Parameters
Xi   = x_Ag(1);
phie = x_Ag(2);
S    = x_Ag(3);
B    = x_Ag(4);
n    = x_Ag(5);
rhoe = x_Ag(6);
```

```
    fe_S = S_Ag*EcFcce_Ag/V;
    rhoe_S = S_Ag*rhoe;

    %  Rho PdPd fitting parameters
    X0_AgAg = [fe_S,Xi,re];
    %  F PdPd fitting parameters
    X1_AgAg = [Fe,rhoe_S,n];
    %  Phi PdPd fitting parametters
    X2_AgAg = [phie,S,B,re];

    X_Phi_PdAg = [X0_PdPd X0_AgAg X2_PdPd X2_AgAg];

        %   F AgAg fitting parameters
    X_F_AgAg   = [Fe,rhoe,n,0,0];
    %   f Pd
    X_f_AgAg   = [fe,Xi,re];
    %   Phi PdPd fitting parametters
    X_Phi_AgAg = [phie,S,B,re];
    %



    % H data
    % F_H
    X_F_HH  =  [aH,bH,cH,dH,rho0H];
    % rho_HH
    X_f_HH   = [CH,DH,0];
    % Phi_HH
    X_phi_HH = [DHH,aHH,bHH,r0HH];


    % PdH data
    % Vectores of fitting parameters
    X_Phi_PdH = [DPdH,aPdH,bPdH,r0PdH];

% % % %  PdAgH untransformed fit 1  f = 0.4417
    X_Phi_AgH = [1.476745339175117   1.967649395899897   1.741864506174084
1.850016663067700];

    % Phi_HH, rho_HH,   F_H
    X_Phi_HH = [X_phi_HH, X_f_HH, X_F_HH];



    phi = {'phi_PdPd';'phi_PdAg';'phi_AgAg';'phi_PdH';'phi_AgH';'phi_HH'};
    f   = {'f_PdPd';'f_AgAg';'f_HH'};
    F   = {'F_Pd';'F_Ag';'F_H'};


    [m n] = size(X_phi_HH);
    X_Phi = zeros(length(phi),n);
    X_Phi(1,1:length(X_Phi_PdPd)) = X_Phi_PdPd;
    X_Phi(2,1:length(X_Phi_PdAg)) = X_Phi_PdAg;
```

```
    X_Phi(3,1:length(X_Phi_AgAg)) = X_Phi_AgAg;
    X_Phi(4,1:length(X_Phi_PdH)) = X_Phi_PdH;
    X_Phi(5,1:length(X_Phi_AgH)) = X_Phi_AgH;
    X_Phi(6,1:length(X_Phi_HH))  = X_Phi_HH;

    X_f   = [X_f_PdPd; X_f_AgAg; X_f_HH];
    X_F   = [X_F_PdPd; X_F_AgAg; X_F_HH];

    element = ['Pd';'Ag';'H '];
    edata   = {46,106.42,3.89,'FCC'; 47,107.87,4.09,'FCC';
1,1.01,3.38,'FCC'};
    %


    rc   = 5.35;
    nr   = 3000;
    nrho = nr;
    dr   = rc/nr;
    drho = 50/nrho;
    r    = [0:dr:nr*dr];
    rho  = [0:drho:drho*nrho];

    alloy = ['Pd';'Ag';'H '];

WritePotential_setf2(alloy,element,edata,nrho,drho,nr,dr,rc,rho,r,f,X_f,F,X_F
,phi,X_Phi)
    fclose(fId);

end


function
WritePotential_setf2(alloy,element,edata,nrho,drho,nr,dr,rc,rho,r,f,X_f,F,X_F
,phi,X_phi)
   global fId;
   fprintf(fId,'setfl format\n');
   s = alloy(1,:);
   for i = 2:length(alloy(:,1))
       s = cat(2,s,alloy(i,:));
   end

   fprintf(fId,'%s functions (universal3)\n',s);
   fprintf(fId,'%s functions (universal3)\n',s);
   fprintf(fId,'%d  %s  %s
%s\n',length(alloy(:,1)),alloy(1,:),alloy(2,:),alloy(3,:));
   fprintf(fId,'%d\t%e\t%d\t%e\t%e\n',nrho,drho,nr,dr,rc);

   for i = 1:length(alloy(:,1))
       for index = 1:length(element)
            if alloy(i,:) == element(index,:)
                %
                Fp = str2func(char(F(index,:)));
                X1 = X_F(index,:);
                EF = Embidding_function_2(rho,Fp,X1);
                %
```

```
                fp = str2func(char(f(index,:)));
                X2 = X_f(index,:);
                Ef = Electron_density_2(r,fp,X2);
                %
                numedata = cell2mat(edata(index,1:3));
                fprintf(fId,'%d\t%6.3f\t%6.3f\t',numedata(1,:));
                structure = string(cell2mat(edata(index,4:end)));
                fprintf(fId,'%s\n',structure);
                %
                Write_F(rho,EF);
                Write_f(r,Ef);
            end
        end
    end



    for index = 1:length(phi)
          phip = str2func(char(phi(index,:)));
          X3 = X_phi(index,:);
          [PHI,RPHI] = Two_body_potential(r,phip,X3);
           Write_rPHI(r,PHI);
    end



end

function FF = Embidding_function_2(rho,F,X)
    for i=1:length(rho)
        if rho(i) ~= 0
            FF(i) = F(rho(i),X);
        else
            FF(i) = 0;
        end
    end
end


function ff = Electron_density_2(r,f,X)
    for i = 1:length(r)
        ff(i) = f(r(i),X);
    end
end


function [PHI,RPHI] = Two_body_potential(r,phi,X)
    for i = 1:length(r)
        PHI(i)  = phi(r(i),X);
        RPHI(i) = r(i)*PHI(i);
    end
end
```

```
%*************************************************************************
%
%*************************************************************************

% Park & Hijazi phi_PdPd
function f = phi_PdPd(rij,X)
    phie = X(1);
    S    = X(2);
    B    = X(3);
    re   = X(4);
    f    = -phie*(1+S*(rij/re-1))*exp(-B*(rij/re-1));
end

%*************************************************************************

% dphidr_H
function f = dphidr_PdPd(r,X)
    phie = X(1);
    S    = X(2);
    B    = X(3);
    re   = X(4);
    f    = (-1/re)*(phie*S*exp(-B*(r/re-1))+B*phi_PdPd(r,X));
end


% d2phidr2_H
function f = d2phidr2_PdPd(r,X)
    phie = X(1);
    S    = X(2);
    B    = X(3);
    re   = X(4);
    f    = (B/re)*((phie*S/re)*exp(-B*(r/re-1))-dphidr_PdPd(r,X));
end


% Electron density
function f = f_PdPd(rij,X)
    fe = X(1);
    Xi = X(2);
    re = X(3);
    f = fe*exp(-Xi*(rij-re));
end

% Park & Hijazi F_P
function f = F_Pd(rho,X)
    f = 0;
    Fe   = X(1);
    rhoe = X(2);
    n    = X(3);
    if rho > 0.0
       f    = -Fe*(1-n*log(rho/rhoe))*(rho/rhoe)^n;
    end
end

function f = phi_HH_u(rij,X)
```

```
    DHH     = X(1);
    alphaHH = X(2);
    betaHH  = X(3);
    r0HH    = X(4);
    f = DHH*(betaHH*exp(-alphaHH*(rij-r0HH)) - alphaHH*exp(-betaHH*(rij-
r0HH)));
end

function f = dphidr_HH_u(r,X)
    DHH     = X(1);
    alphaHH = X(2);
    betaHH  = X(3);
    r0HH    = X(4);
    f = DHH*( -alphaHH * betaHH*exp(-alphaHH*(r-r0HH)) + betaHH*alphaHH*exp(-
betaHH*(r-r0HH)) );
end

%
function f = d2phidr2_HH_u(r,X)
     DHH     = X(1);
     alphaHH = X(2);
     betaHH  = X(3);
     r0HH    = X(4);
     f = DHH*( (alphaHH^2) * betaHH*exp(-alphaHH*(r-r0HH)) -
(betaHH^2)*alphaHH*exp(-betaHH*(r-r0HH)) );
end

% % H-H normalized pair potential as a function of interatomic distance rij
function f = phi_HH(rij,X)
    X2_HH = X(1:1:4);
    X0_HH = X(5:1:7);
    X1_HH = X(8:1:12);
    rho0H = X(12);
    f = phi_HH_u(rij,X2_HH) + 2 * dFdrho_H_u(rho0H,X1_HH) * f_HH(rij,X0_HH);
end

% % First derivative of H-H normalized pair potential with respect to rij
function f = dphidr_HH(rij,X)
    X2_HH = X(1:1:4);
    X0_HH = X(5:1:7);
    X1_HH = X(8:1:12);
    rho0H = X(12);
    f = dphidr_HH_u(rij,X2_HH) + 2 * dFdrho_H_u(rho0H,X1_HH) * ...
        dfdr_HH(rij,X0_HH);
end

% % Second derivative of H-H normalized pair potential with respect to rij
function f = d2phidr2_HH(rij,X)
    X2_HH = X(1:1:4);
    X0_HH = X(5:1:7);
    X1_HH = X(8:1:12);
    rho0H = X(12);
    f = d2phidr2_HH_u(rij,X2_HH) + 2 * dFdrho_H_u(rho0H,X1_HH) * ...
        d2fdr2_HH(rij,X0_HH);
end
```

```
% H Atomic Electron Density as a function of interatomic distance
function f = f_HH(r,X)
    CH = X(1);
    DH = X(2);
    f = CH*exp(-DH*r);
end

% First derivative H Atomic Electron Density with respect to rij
function f = dfdr_HH(r,X)
    CH = X(1);
    DH = X(2);
    f = -DH*CH*exp(-DH*r);
end

% Second derivative H Atomic Electron Density with respect to rij
function f = d2fdr2_HH(r,X)
    CH = X(1);
    DH = X(2);
    f = (DH^2)*CH*exp(-DH*r);
end

% H unormalized Embedding Energy function
function f = F_H_u(rho,X)
    aH = X(1);
    bH = X(2);
    cH = X(3);
    dH = X(4);
    EH = 0.0540638;
    f = -cH * ( (1/(2+dH))*(rho+EH)^(2+dH) - ((aH + bH)/(1+dH))*...
        (rho+EH)^(1+dH) + ((aH * bH)/dH) *(rho+EH)^dH );
end

% First derivative of H unnormalized Embedding function with respect to Rho
function f = dFdrho_H_u(rho,X)
    aH = X(1);
    bH = X(2);
    cH = X(3);
    dH = X(4);
    EH = 0.0540638;
    f = -cH * ( (rho+EH)^(1+dH) - (aH + bH) * (rho+EH)^(dH) +...
        (aH * bH) *(rho+EH)^(dH-1) );
end

% Second derivative of H unnormalized Embedding function with respect to Rho
function f = d2Fdrho2_H_u(rho,X)
    aH = X(1);
    bH = X(2);
    cH = X(3);
    dH = X(4);
    EH = 0.0540638;
    f = -cH * ( (1+dH)*(rho+EH)^(dH) - dH*(aH + bH) * (rho+EH)^(dH-1) + ...
        (dH-1)*(aH * bH) *(rho+EH)^(dH-2) );
end

% H normalized Embedding Energy function
function f = F_H(rho,X)
```

```
    X1_HH = X(1:1:4);
    rho0H = X(5);
    f = F_H_u(rho,X1_HH) - dFdrho_H_u(rho0H,X1_HH) * rho;
end

% First derivative of H normalized Embedding function with respect to Rho
function f = dFdrho_H(rho,X)
    X1_HH = X(1:1:4);
    rho0H = X(5);
    f = dFdrho_H_u(rho,X1_HH) - dFdrho_H_u(rho0H,X1_HH);
end

% Second derivative of H normalized Embedding function with respect to Rho
function f = d2Fdrho2_H(rho,X)
    X1_HH = X(1:1:4);
    f = d2Fdrho2_H_u(rho,X1_HH);
end

% Pd-H pair potential as a function of interatomic destance rij
function f = phi_PdH(r,X)
    DPdH      = X(1);
    alphaPdH  = X(2);
    betaPdH   = X(3);
    r0PdH     = X(4);
    f = DPdH*( betaPdH*exp(-alphaPdH*(r-r0PdH)) - ...
        alphaPdH*exp(-betaPdH*(r-r0PdH)) );
end

% First derivative of Pd-H pair potential with respect to interatomic
% destance rij
function f = dphidr_PdH(r,X)
        DPdH      = X(1);
        alphaPdH  = X(2);
        betaPdH   = X(3);
        r0PdH     = X(4);
        f = DPdH*( -alphaPdH*betaPdH*exp(-alphaPdH*(r-r0PdH)) +...
            betaPdH*alphaPdH*exp(-betaPdH*(r-r0PdH)) );
end

% Second derivative of Pd-H pair potential with respect to interatomic
% destance rij
function f = d2phidr2_PdH(r,X)
        DPdH      = X(1);
        alphaPdH  = X(2);
        betaPdH   = X(3);
        r0PdH     = X(4);
        f = DPdH*( (alphaPdH^2)*betaPdH*exp(-alphaPdH*(r-r0PdH)) - ...
            (betaPdH^2)*alphaPdH*exp(-betaPdH*(r-r0PdH)) );
end

%*************************************************************************
%
%*************************************************************************
function [afcce,Ecfcce,c11e,c12e,c44e,Bme,Eve,abcce,Ecbcce,cpre] =
parameters(el)
    index = 1;
```

```
    elements = ['Ag';'Al';'Au';'Cu';'Ni';'Pd';'Pt';'Hi'];
    for i = 1:length(elements)
        x = elements(i,:);
        if el == x
          index = i;
        end
    end
    afcc =  [4.09;4.05;4.08;3.615;3.52;3.89;3.92;3.38];
    Ecfcc = [2.85;3.36;3.93;3.54;4.45;3.91;5.77;2.119];
    c11 =   [1.24;1.14;1.86;1.70;2.465;2.341;3.47;0.0];
    c12 =   [0.934;0.619;1.57;1.225;1.473;1.761;2.51;0.0];
    c44 =   [0.461;0.316;0.42;0.758;1.247;0.712;0.765;0.05];
    cpr =   [1,1,1,1,1,1,1,0.05];
    % Unrelaxed
    Evfcc = [1.1;0.866;0.9;1.3;1.70;1.54;1.60;0.0];
    % Relaxed
%   Evfcc = [1.3;1.1;0.75;0.9;1.60;1.40;1.50;0.0];
    B =     [1.04;0.793;1.67;1.38;1.804;1.95;2.83;0.50];
    abcc =  [3.32;3.3;3.18;2.87;2.89;3.16;3.21;0.0];
    Ecbcc = [1;1;1;3.49;1;1;1;1];

    %
    afcce = afcc(index);
    Ecfcce = Ecfcc(index);
    Eve = Evfcc(index);
    c11e = c11(index);
    c12e = c12(index);
    c44e = c44(index);
    Bme = B(index);
    abcce = abcc(index);
    Ecbcce = Ecbcc(index);
    cpre   = cpr(index);
end

%
function [afcce,Ecfcce] = PdH_parameters(el)
    index1 = 1;
    elements =
['Pd000H';'PdH000';'PdH025';'PdH050';'PdH075';'PdH100';'PdHT50';'PdHT75'];
    [n,m]  = size(elements);
    for i = 1:n
        x = elements(i,:);
        if el == x
          index1 = i;
        end
    end

    % From Sandia Paper
    a  =  [3.38;3.89;3.9421;4.0007;4.0593;4.1179;4.0007;4.0593];
    Ec =  [2.119;3.91;3.64024;3.48770;3.37031;3.26966;3.50000;3.50000];
    %
    afcce  = a(index1);
    Ecfcce = Ec(index1);
end

%
```

```
function Write_F(rho,F)
    global fId;
    %Write F
    for i = 1:5:length(rho)-1
        for j = 0:4
            fprintf(fId,'%20.16e\t',F(i+j));
%             fprintf(fId,'%f\t',F(i+j));
        end
            fprintf(fId,'\n');
   end
end

%
function Write_f(r,f)
    global fId;
    %Write f
    for i = 1:5
        f(length(r)-i)=0;
    end
    for i = 1:5:length(r)-1
        for j = 0:4
            fprintf(fId,'%20.16e\t',f(i+j));
%             fprintf(fId,'%f\t',f(i+j));
        end
        fprintf(fId,'\n');
    end
end

%
function Write_rPHI(r,PHI)
   global fId;
   %Write PHI
   for i = 1:5
        PHI(length(r)-i)=0;
   end
   %k = 1;
   for i = 1:5:length(r)-1
        for j = 0:4
           fprintf(fId,'%20.16e\t',r(i+j)*PHI(i+j));
%             fprintf(fId,'%f\t',r(i+j)*PHI(i+j));
        end
        fprintf(fId,'\n');
   end
end

function f = phi_PdAg(rij,X)
     X0_a = X(1:3);
     X0_b = X(4:6);
     X2_a = X(7:10);
     X2_b = X(11:14);
     f_a = f_PdPd(rij,X0_a);
     f_b = f_PdPd(rij,X0_b);
     phi_a = phi_PdPd(rij,X2_a);
     phi_b = phi_PdPd(rij,X2_b);
     f = .5*( f_b*f_a^-1*phi_a + f_a*f_b^-1*phi_b );
end
```

```
function f = phi_AgH(r,X)
    DPdH      = X(1);
    alphaPdH  = X(2);
    betaPdH   = X(3);
    r0PdH     = X(4);
    f = DPdH*( betaPdH*exp(-alphaPdH*(r-r0PdH)) - ...
        alphaPdH*exp(-betaPdH*(r-r0PdH)) );
end

% Electron density
function f = f_AgAg(rij,X)
    fe = X(1);
    Xi = X(2);
    re = X(3);
    f = fe*exp(-Xi*(rij-re));
end

% Park & Hijazi F_P
function f = F_Ag(rho,X)
    f = 0;
    Fe   = X(1);
    rhoe = X(2);
    n    = X(3);
    if rho > 0.0
       f    = -Fe*(1-n*log(rho/rhoe))*(rho/rhoe)^n;
    end
end

% Park & Hijazi phi_AgAg
function f = phi_AgAg(rij,X)
    phie = X(1);
    S    = X(2);
    B    = X(3);
    re   = X(4);
    f    = -phie*(1+S*(rij/re-1))*exp(-B*(rij/re-1));
end
```